University of Sussex

Novel Mathematical and Computational Approaches for Modelling Biological Systems

Andy Heung Wing Chung

Submitted for the degree of Doctor of Philosophy

University of Sussex

March 2016

Declaration

I hereby declare that this thesis has not been and will not be submitted in whole or in part to another University for the award of any other degree.

Signature:

Andy Heung Wing Chung

UNIVERSITY OF SUSSEX

Andy Heung Wing Chung, Doctor of Philosophy

Novel Mathematical and Computational Approaches for Modelling Biological Systems

Summary

This work presents the development, analysis and subsequent simulations of mathematical models aimed at providing a basis for modelling atherosclerosis. This cardiovascular disease is characterised by the growth of plaque in artery walls, forming lesions that protrude into the lumen. The rupture of these lesions contributes greatly to the number of cases of stroke and myocardial infarction. These are two of the main causes of death in the UK. Any work to understand the processes by which the disease initiates and progresses has the ultimate aim of limiting the disease through either its prevention or medical treatment and thus contributes a relevant addition to the growing body of research.

The literature supports the view that the cause of atherosclerotic lesions is an inflammatory process - succinctly put, excess amounts of certain biochemical species fed into the artery wall via the bloodstream spur the focal accumulation of extraneous cells. Therefore, suitable components of a mathematical model would include descriptions of the interactions of the various biochemical species and their movement in space and time.

The models considered here are in the form of partial differential equations. Specifically, the following models are examined: first, a system of reaction-diffusion equations with coupling between surface and bulk species; second, a problem of optimisation to identify an unknown boundary; and finally, a system of advection-reaction-diffusion equations to model the assembly of keratin networks inside cells. These equations are approximated and solved computationally using the finite element method. The methods and algorithms shown aim to provide more accurate and efficient means to obtain solutions to such equations.

Each model in this work is extensible and with elements from each model combined, they have scope to be a platform to give a fuller model of atherosclerosis.

Acknowledgements

I give great thanks and am grateful to Anotida Madzvamuse and Omar Lakkis who, as my supervisors, guided and encouraged me throughout this project. I thank Chandrasekhar Venkataraman, Erik Burman, Rachid Touzani and François Bouchon whose collaboration made the work presented in this thesis possible. My thanks to Stéphanie Portet, Rudolf Leube, Reinhard Windhoffer whose combined efforts brought about fruitful research. My thanks also to Abdullah Jibawi and Toby Richards for their valued partnership. I would also like to thank Vanessa Styles and Philip Maini for their helpful input.

This project was funded jointly by the University of Sussex and the Medical Research Council who have my lasting gratitude.

Contents

List of Tables ix

List of Figures 1

1 Introduction 2

1.1 Circulatory System . . . 5

1.2 Immune Response . . . 8

1.3 Cholesterol metabolism . . . 11

1.4 The Onset of Atherosclerosis: atherogenesis . . . 12

1.5 The Progression of Atherosclerosis and Clinical Relevance . . . 14

1.6 Atherosclerosis in the clinic . . . 18

1.6.1 Preventative measures . . . 18

1.6.2 Diagnosis . . . 18

1.6.3 Treatment . . . 19

1.7 Mathematical Models of Atherosclerosis . . . 20

1.8 Summary . . . 25

2 Numerical Methods for Reaction-Diffusion Systems 27

2.1 The model equations . . . 28

2.2 Numerical methods . . . 30

2.2.1 Weak Formulation . . . 31

2.2.2 Finite element spatial discretisation . . . 31

2.2.3 Space discretisation . . . 31

2.2.4 Time discretisation . . . 32

2.2.5 Techniques for treating the non-linearities . . . 33

2.3 Computation Implementation . . . 37

2.4 Experimental order of convergence (EOC) . . . 40
2.5 Numerical Results . . . 45
2.6 Comparison with IMEX schemes . . . 51
2.7 Applications to other geometries . . . 54
2.8 Conclusions . . . 56

3 Bulk-surface coupled RDEs **58**
3.1 Introduction . . . 59
3.2 Coupled bulk-surface reaction-diffusion systems on stationary volumes . . . 61
3.2.1 A coupled system of bulk-surface reaction-diffusion equations (BSRDEs) . . . 62
3.2.2 Linear stability analysis of the coupled system of BSRDEs . . . 64
3.2.3 Weak variational form . . . 76
3.2.4 Discretisation . . . 76
3.3 Computation Implementation . . . 79
3.4 Numerical simulations of the coupled system of bulk-surface reaction-diffusion equations (BSRDEs) . . . 81
3.4.1 Investigating the boundary behaviour . . . 83
3.5 Weak vs. Strong Coupling . . . 93
3.6 Conclusion, discussion and future research challenges . . . 96

4 A Shape Identification Problem **98**
4.1 Introduction . . . 98
4.2 Statement of the Problem . . . 99
4.2.1 Basic notation . . . 99
4.2.2 The axially symmetric problem . . . 99
4.2.3 An overview of the approach . . . 101
4.2.4 Turning a free boundary problem into a free coefficients PDE . . . 101
4.2.5 A fixed boundary formulation of Problem (4.2.2) . . . 104
4.3 Constrained optimization and regularization . . . 104
4.3.1 Definitions . . . 104
4.3.2 Optimality system . . . 105
4.4 A descent method . . . 106

4.4.1 An iterative descent algorithm 106
4.5 Computational Implementation 107
4.6 Numerical results 108
4.7 Conclusions 120

5 Conclusion and further work **121**

Appendices **123**

A **124**

A.1 Exploring the Turing Space 124

B **128**

List of Tables

1.1 AHA classification of atherosclerosis as laid out in (Stary et al. 1994, 1995) 17

2.1 Experimental order of convergence (EOC) for the u-variable. Further EOC values were computed for BE to investigate the convergence to the expected value of 1. 42

2.2 Details of the simulation of the Schnakenberg system using (a) Picard iteration and (b) the Newton method. Shown are the total number of non-linear iterations required to reach the end time and the elapsed CPU time. Convergence was not obtained with the CN method, therefore the data is not included. 46

2.3 Details of the simulation of the Schnakenberg system using the IMEX schemes with one Picard iteration. Shown are the end time of the simulations and the elapsed CPU time. Convergence was not reached with either the CIMEX or C5IMEX methods, therefore the data is not included. 51

2.4 Details of the simulation of the Schnakenberg system using the one Newton iteration. Convergence was not obtained with the CN method, therefore the data in not included. 52

3.1 Model parameter values for the coupled system of BSRDEs (3.2.1) - (3.2.4). 81

3.2 Model parameter values used in simulations for Figures 3.3-3.9. 82

3.3 Model parameter values for the coupled system of BSRDEs (3.2.1) - (3.2.4), where ε is some non-negative parameter. 93

3.4 Model parameter values used in simulations for Figures 3.11-3.16 93

4.1 Results on a 64×64 grid (4096 nodes) with $\mu_1 = 0$ with $s_{ex}(t) = 1 - 8t^2(1-t)^2$. Graphs shown in Fig. B.1 111

4.2 Results on a 64×64 grid (4096 nodes) with $\mu_1 = 0.5$ with $s_{ex}(t) = 1 - 8t^2(1-t)^2$. Graphs shown in Fig. B.1 112

4.3 Results on a 64×64 grid (4096 nodes) with $\mu_1 = 1$ with $s_{ex}(t) = 1 - 8t^2(1-t)^2$. Graphs shown in Fig. B.1 113

4.4 Results on a 64×64 grid (4096 nodes) with $\mu_1 = 0$ with $s_{ex}(t) = 1 - 5t^3(1-t)$. Graphs shown in Fig. B.7 114

4.5 Results on a 64×64 grid (4096 nodes) with $\mu_1 = 0.5$ with $s_{ex}(t) = 1 - 5t^3(1-t)$. Graphs shown in Fig. B.7 115

4.6 Results on a 64×64 grid (4096 nodes) with $\mu_1 = 1$ with $s_{ex}(t) = 1 - 5t^3(1-t)$. Graphs shown in Fig. B.7 116

4.7 Results on a 64×64 grid (4096 nodes) with $\mu_1 = 0$ with $s_{ex}(t) = 1 - 8t^2(1-t)^2$. Graphs shown in Fig. B.13 117

4.8 Results on a 64×64 grid (4096 nodes) with $\mu_1 = 0.5$ with $s_{ex}(t) = (1 + |2t-1|)/2$. Graphs shown in Fig. B.13 118

4.9 Results on a 64×64 grid (4096 nodes) with $\mu_1 = 1$ with $s_{ex}(t) = (1 + |2t-1|)/2$. Graphs shown in Fig. B.13 119

List of Figures

1.1 Structure of an artery 8

1.2 Basic structure of an antibody: (1) Fab region; (2) Fc region; (3) light chain; (4) heavy chain; (5) disulphide bonds; (6), (7) light chain constant and variable domains resp.; (8) heavy chain variable domain; (9), (10), (11) heavy chain constant domains. 9

1.3 Typical 2D computational geometry of an artery segment. For clarity, the top wall has not been labelled. 22

2.1 A simple example of a mesh of a rectangle using quadrilateral elements. The elements have been lettered *a*-*d* and the vertices have been numbered 1-9. 39

2.2 Errors for the modified equation for the u-variable using the adaptive Newton method: (a) BE (b) CN (c) FTS. The second order methods CN and FSTS are more accurate than BE. The error ofconvergence results are shown in Table 2.1. 43

2.3 Errors in the simulation of the modified equation (2.4.2) using the modified CN methods CNB, CNB2 and CNB5 with (a) $\tau = 1/2$ and (b) $\tau = 1/32$ (see Section 2.4 for details). The oscillations observed in CN are damped by the modified methods for both these timesteps. An adaptive Newton method was employed for the fully implicit time-stepping scheme. 44

2.4 Number of non-linear iterations for (a) Picard iteration, n_p, and (b) the Newton method, n_n, at each timestep. 47

2.5 Convergence history of the simulation of the Schnakenberg system for (a) the u-variable and (b) the v-variable. Seen is the initial decay of the modes, mode excitation (growth) and decay into the inhomogeneous steady state. An adaptive Newton method was employed for the fully implicit time-stepping scheme. 48

2.6 Comparison between the convergence history of the FSTS and the modified CNB5 method for the u-variable. Much better agreement is seen here compared to the unmodified CN method. Though not shown, similar improvement is seen in the v-variabe as well. An adaptive Newton method was employed for the fully implicit time-stepping scheme. 49

2.7 A comparison of growth rates - the horizontal lines correspond to the theoretical growth rate as in the RHS of (2.5.2). (a) BE; (b) CN, CNB5 and FSTS. Excellent agreement during the period of mode excitation is seen between the theoretical growth rate and that obtained in the numerical simulations using BE, CNB5 and FSTS. Using CN, this agreement is lost. An adaptive Newton method was employed for the fully implicit time-stepping scheme. 50

2.8 Comparison of the convergence history of the simulation of the Schnakenberg system for the IMEX schemes. The solid lines show the IMEX schemes and the dotted lines their fully implicit counterparts. 53

2.9 Solution for the variable u of the Schnakenberg system on the square (top row), the bulk of the cube (middle row) and the bulk of the sphere (bottom row). Parameter values were $a = 0.1$, $b = 0.9$ and (a), (c), (e) $d = 9.1676$, $\gamma = 176.72$ and (b), (d), (f) $d = 8.6076$, $\gamma = 535.09$. Part of the domain has been cut away and shown on the right in the 3D geometries to reveal some internal structure. A single Newton iteration at each timestep was employed for the fully implicit time-stepping scheme. 55

3.1 The variables of the bulk-surface coupled system. On the left is shown the whole bulk domain Ω. We make a cut along the dotted line and show an exploded view on the right. The variables u and v are defined both in the interior and boundary of Ω, while the variables r and s are only defined on the boundary. 62

3.2 Example meshes for the bulk (top) and surface system (bottom). Part of the domain has been cut away and shown on the right to reveal some internal mesh structure. 84

3.3 Numerical solutions corresponding to the coupled system of BSRDEs (3.2.1)-(3.2.5) with $d_\Omega = 1$ in the bulk, $d_\Gamma = 1$ on the surface and $\gamma_\Omega = \gamma_\Gamma = 500$. The uniform steady state solutions are converged to and no patterns form. Rows 1, 3, and 5: solutions in the bulk representing u and v. Rows 2, 4 and 6: solutions on the surface representing r and s. Second and fourth columns represent cross sections of the bulk and the surface respectively. . 85

3.4 Numerical solutions corresponding to the coupled system of BSRDEs (3.2.1)-(3.2.5) with $d_\Omega = 1$ in the bulk, $d_\Gamma = 10$ on the surface and $\gamma_\Omega = \gamma_\Gamma = 500$. The surface reaction-diffusion system induces patterning in a small band close to the surface. In the bulk, no patterns form almost everywhere. The patterning behaviour is independent of the geometry. Rows 1, 3, and 5: solutions in the bulk representing u and v. Rows 2, 4 and 6: solutions on the surface representing r and s. Second and fourth columns represent cross sections of the bulk and the surface respectively. 86

3.5 Numerical solutions corresponding to the coupled system of BSRDEs (3.2.1)-(3.2.5) with $d_\Omega = 1$ in the bulk, $d_\Gamma = 20$ on the surface and $\gamma_\Omega = \gamma_\Gamma = 500$. The pattering process is similar to that shown in Figure 3.4 for large values of d_Γ. Rows 2, 4 and 6: solutions on the surface representing r and s. Second and fourth columns represent cross sections of the bulk and the surface respectively. 87

3.6 Numerical solutions corresponding to the coupled system of BSRDEs (3.2.1)-(3.2.5) with $d_\Omega = 10$ in the bulk, $d_\Gamma = 1$ on the surface and $\gamma_\Omega = \gamma_\Gamma = 500$. The bulk reaction-diffusion system is able to induce patterning almost everywhere on the surface. Rows 2, 4 and 6: solutions on the surface representing r and s. Second and fourth columns represent cross sections of the bulk and the surface respectively. 88

3.7 Numerical solutions corresponding to the coupled system of BSRDEs (3.2.1)-(3.2.5) with $d_\Omega = 10$ in the bulk, $d_\Gamma = 10$ on the surface and $\gamma_\Omega = \gamma_\Gamma = 500$. The bulk and surface reaction-diffusion systems both have the capability to induce patterning. Rows 2, 4 and 6: solutions on the surface representing r and s. Second and fourth columns represent cross sections of the bulk and the surface respectively. 89

3.8 Numerical solutions corresponding to the coupled system of BSRDEs (3.2.1)-(3.2.5) with $d_\Omega = 20$ in the bulk, $d_\Gamma = 10$ on the surface and $\gamma_\Omega = \gamma_\Gamma = 500$. The coupled system of BSRDEs induces pattering in the bulk and on the surface. Rows 2, 4 and 6: solutions on the surface representing r and s. Second and fourth columns represent cross sections of the bulk and the surface respectively. 90

3.9 Numerical solutions corresponding to the coupled system of BSRDEs (3.2.1)-(3.2.5) with $d_\Omega = 20$ in the bulk, $d_\Gamma = 20$ on the surface and $\gamma_\Omega = \gamma_\Gamma = 500$. Patterning behaviour is more complex for large values of the diffusion coefficients. Rows 2, 4 and 6: solutions on the surface representing r and s. Second and fourth columns represent cross sections of the bulk and the surface, respectively. 91

3.10 Line plots of the solution u for the simulations in the unit sphere centred on the origin with varying values of d_Ω and d_Γ. The solution is plotted along the straight line extending from the origin to: (a) a point on the sphere with the maximum boundary value of u; (b) the point $(1, 0, 0)$. The two values in the legends correspond to the values (d_Ω, d_Γ) in that order. The top graph holds the value $d_\Omega = 1$ fixed whilst the bottom graph holds the value $d_\Gamma = 1$ fixed. The value of 0 on the horizontal axis corresponds to the origin and the value of 1 corresponds to a point on the surface of the spherical domain. 92

3.11 Numerical solution to the coupled system of BSRDEs (3.2.1) - (3.2.4) using the parameter values shown in Table 3.3 with $\varepsilon = 0$ 94

3.12 Numerical solution to the coupled system of BSRDEs (3.2.1) - (3.2.4) using the parameter values shown in Table 3.3 with $\varepsilon = 0.001$ 94

3.13 Numerical solution to the coupled system of BSRDEs (3.2.1) - (3.2.4) using the parameter values shown in Table 3.3 with $\varepsilon = 0.01$ 94
3.14 Numerical solution to the coupled system of BSRDEs (3.2.1) - (3.2.4) using the parameter values shown in Table 3.3 with $\varepsilon = 0.1$ 95
3.15 Numerical solution to the coupled system of BSRDEs (3.2.1) - (3.2.4) using the parameter values shown in Table 3.3 with $\varepsilon = 1$ 95
3.16 Numerical solution to the coupled system of BSRDEs (3.2.1) - (3.2.4) using the parameter values shown in Table 3.3 with $\varepsilon = 10$ 95

4.1 The test-cases $s_{ex}(t)$ used for testing the algorithm. 109
4.2 Initial guesses $s_0(t)$ used to start the algorithm (see equation (4.6.6)). . . . 110

A.1 The Turing space for the Schakenberg equation for different values of d. The coloured area shows the region where all the conditions (A.1.7)-(A.1.10) are all satisfied. 126
A.2 The mesh used to calculate the numerical results shown in Fig. A.3. 126
A.3 Numerical solutions of the Schnakenberg system with $d = 50$, $a = 0.1$ and $\gamma = 29$ for diffrent values of b. 127

B.1 Results for $s_0(t) = 1$ and $s_{ex}(t) = 1 - 8t^2(1-t)^2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.1. 129
B.2 Results for $s_0(t) = 1 + 0.2t(1-t)\sin(8\pi t)$ and $s_{ex}(t) = 1 - 8t^2(1-t)^2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.1. 130
B.3 Results for $s_0(t) = 0.5 + 0.5 s_{ex}(t)$ and $s_{ex}(t) = 1 - 8t^2(1-t)^2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.2. . . . 131
B.4 Results for $s_0(t) = 0.5 + 0.5 s_{ex}(t) + 0.2t(1-t)\sin(8pi * t)$ and $s_{ex}(t) = 1 - 8t^2(1-t)^2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.2. 132
B.5 Results for $s_0(t) = s_{ex}(t)$ and $s_{ex}(t) = 1 - 8t^2(1-t)^2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.3. 133
B.6 Results for $s_0(t) = s_{ex}(t) + 0.2t(1-t)\sin(8\pi t)$ and $s_{ex}(t) = 1 - 8t^2(1-t)^2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.3. 134

B.7 Results for $s_0(t) = 1$ and $s_{ex}(t) = 1 - 5t^3(1-t)$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.4. 135

B.8 Results for $s_0(t) = 1 + 0.2t(1-t)\sin(8\pi t)$ and $s_{ex}(t) = 1 - 5t^3(1-t)$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.4. 136

B.9 Results for $s_0(t) = 0.5 + 0.5s_{ex}(t)$ and $s_{ex}(t) = 1 - 5t^3(1-t)$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.5. . . . 137

B.10 Results for $s_0(t) = 0.5 + 0.5s_{ex}(t) + 0.2t(1-t)\sin(8pi * t)$ and $s_{ex}(t) = 1 - 5t^3(1-t)$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.5. 138

B.11 Results for $s_0(t) = s_{ex}(t)$ and $s_{ex}(t) = 1 - 5t^3(1-t)$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.6. 139

B.12 Results for $s_0(t) = s_{ex}(t) + 0.2t(1-t)\sin(8\pi t)$ and $s_{ex}(t) = 1 - 5t^3(1-t)$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.6. 140

B.13 Results for $s_0(t) = 1$ and $s_{ex}(t) = (1 + |2t-1|)/2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.7. 141

B.14 Results for $s_0(t) = 1 + 0.2t(1-t)\sin(8\pi t)$ and $s_{ex}(t) = (1 + |2t-1|)/2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.7. 142

B.15 Results for $s_0(t) = 0.5 + 0.5s_{ex}(t)$ and $s_{ex}(t) = (1 + |2t-1|)/2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.8. . . . 143

B.16 Results for $s_0(t) = 0.5 + 0.5s_{ex}(t) + 0.2t(1-t)\sin(8pi * t)$ and $s_{ex}(t) = (1 + |2t-1|)/2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.8. 144

B.17 Results for $s_0(t) = s_{ex}(t)$ and $s_{ex}(t) = (1 + |2t-1|)/2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.9. 145

B.18 Results for $s_0(t) = s_{ex}(t) + 0.2t(1-t)\sin(8\pi t)$ and $s_{ex}(t) = (1 + |2t-1|)/2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.9. 146

Chapter 1

Introduction

In this thesis, we aim to develop novel and efficient computational approaches to solving coupled systems of reaction-diffusion equations and inverse problems. These types of problems are ubiquitous in the modelling of biological systems: the applications of reaction-diffusion systems include developmental biology, cell motility, cancer biology, astrophysics, semiconductor physics, ecology, material science, chemistry, financial mathematics and textile engineering (Ascher et al. 1995, Atkinson 1989, Barreira et al. 2011, Batchelor 2000, Britz et al. 2003, Castets et al. 1990, Dufiet & Boissonade 1992, Madzvamuse et al. 2003, Murray 2003, Østerby 2003) while inverse problems find applications in fields as diverse as medicine, hydrodynamics, food science, geoscience, financial engineering and reaction-diffusion systems (Cañavate-Grimal et al. 2015, Fabbri & Cevoli 2016, Gnanavel et al. 2013, Jiang & Tao 2001, Nandaa et al. 2014, Yanga & Hamrick 2005). Thus, it is vital that the methods used to solve these types of problems numerically are both accurate and efficient.

Since the pioneering work of Turing (Turing 1952), a wide variety of models of reaction-diffusion equations (RDEs) have been proposed as plausible mechanisms for pattern generation processes (Murray 2003). Turing derived the conditions under which a linearised reaction-diffusion system admits a linearly stable spatially homogeneous steady state in the absence of diffusion and yet, in the presence of diffusion, it becomes unstable under appropriate conditions to yield a spatially varying inhomogeneous steady state. This process is now well-known as diffusively-driven instability and is of particular interest in developmental biological pattern formation as a means of initiating self organisation from a virtually homogeneous background. Turing patterns were first observed by Castets *et.*

al. (Castets et al. 1990) in a chloride-ionic-malonic-acid reaction. Ouyang and Swinney (Ouyang & Swinney 1991) were the first to observe a Turing instability from a spatially uniform state to a patterned state. Although controversial in a biological context for many years, recent experimental findings suggest this may be a mechanism for the formation of repeated structures in skin organ formation (Sick et al. 2006, Maini et al. 2006) and zebrafish mesoderm cell fates (Solnica-Krezel 2003). Beyond developmental biology, we have seen above that reaction-diffusion models are widely used a variety of applications.

In many cases, these models are comprised of highly nonlinear reaction terms which makes it impossible to obtain analytical solutions in closed form. Hence numerical methods are employed. For the reaction-diffusion theory of pattern formation, these methods play two key roles: (i) they are used to validate linear stability theoretical results close to bifurcation points (and vice versa) and (ii) far away from bifurcation points, they provide numerical approximate solutions (in the absence of analytical solutions). A typical numerical method consists of two processes: first, a space discretisation is applied to render the system of partial differential equations (PDEs) into a system of ordinary differentials equations (ODEs) and second, a time discretisation is employed, thereby transforming the system of ODEs into a system of linear or nonlinear algebraic equations depending on the form of the time discretisation scheme. Finally, techniques from numerical linear algebra are employed to solve efficiently the resulting system.

Space discretisation methods include (but are not limited to) finite differences, finite elements, spectral methods, finite volume, closest point methods on stationary domains, volumes and surfaces and more recently moving grid, surface finite element and particle methods applied to domains and surfaces that evolve in time (Madzvamuse et al. 2003, Barreira et al. 2011). In this thesis, we choose to use the finite element methodology since it can cope easily with complex irregular geometries, volumes and surfaces. Other numerical methods can be applied, however, for some methods (e.g. finite differences) their extension to complicated, irregular and sometimes continuously evolving domains and surfaces is not at all trivial. The applicability of finite element methods to complicated domains is well known. The finite element method can easily deal with complicated and sometimes continuously changing domains.

Several time discretisation schemes have been widely used to compute solutions of PDEs on stationary and evolving domains and surfaces (Madzvamuse 2006). These in-

clude the forward Euler method (the most commonly used in computational biology), Gear's method, a modified Euler predictor-corrector method, Gourlay's method (Dufiet & Boissonade 1992), a semi-implicit Rosenbock integrator (Dufiet & Boissonade 1992), and Runge-Kutta schemes (Hairer et al. 1987). Most of these are inadequate because of the stiffness of the diffusive term. Fully explicit methods require excessively small time-steps resulting in computations that are prohibitively expensive especially in multi-dimensions. Recently, IMEX schemes have been used to solve RDEs on stationary one-dimensional domains (Ascher et al. 1995, Ruuth 1995). The key essence of these schemes is that an implicit scheme is applied to approximate the diffusive term and an explicit scheme is used to approximate the reaction kinetics, hence the name IMEX. More recently, Madzvamuse (Madzvamuse 2006) presented several time-stepping schemes to compute solutions to reaction-diffusion systems on fixed and growing domains. A first order semi-implicit backward Euler differentiation formula (1-SBEM) which treats diffusive and linear reaction terms implicitly and nonlinear reaction terms semi-implicitly was introduced and shown to be more robust than IMEX schemes. The 1-SBEM employed a single Picard iteration. In all these studies, very little work has been done to extend such analysis to fully implicit schemes. Therefore in this thesis we substantially extend previous IMEX schemes to fully implicit time-stepping schemes for RDEs on stationary domains and surfaces. Fully implicit time-stepping schemes offer greater numerical stability than the 1-SBEM and IMEX schemes.

Inverse problems represent a class of problems that seeks parameters for a given model given some set of observed data. This is in contrast to the more straightforward case where a model is given with a set of parameters and is solved to get a set of simulated data. Typically, a model is built which depends on some parameters and an error function is defined which measures the 'distance' from the observed data to data calculated by solving the model. The problem is then an optimisation problem to minimise the error. Many optimisation techniques can be used to attempt a solution to the problem, such as gradient descent, Newton's method, the Nelder-Mead method or the simplex algorithm (see for example Nocedal & Wright (2006)).

Inverse problems are essential when a physical model may be built for a particular system but the parameters used within it are not known. The retrieval of the parameters may provide helpful insight to the problem at hand. Inverse problems are often very

difficult to solve as they are often ill-posed and are made more difficult by the fact that the observed data used as input often contains errors in measurement due to physical constraints (for example see Isakov (2005)). There are thus issues with uniqueness and existence of solutions to these problems making numerical solutions difficult to compute. Regularisation is employed to stabilise the problem. A commonly used form of regularisation is Tikhonov regularisation, first described by Tikhonov in (Tikhonov 1943) (or see (Tikhonov & Arsenin 1977) for an English translation).

Alternatively, a probabilistic approach may be used leading to a maximal likelihood approach or a Bayesian approach to inverse problems (see for example Kaipio & Somersalo (2004) and Idier (2008)). Probabilistic approaches have the benefit of being able to model and quantify the uncertainties in the data and parameters. This information is valuable when interpreting the results of the resulting model. In the area of biology and medicine, which is the example case in this thesis, both the deterministic and probabilistic approaches have found use in the literature (Clermonta & Zenke 2015, Zenker et al. 2007, Klinke & Birtwistle 2015). In this thesis, we shall use a deterministic approach to solve a shape identification problem.

We shall focus on numerical and computational methods used to solve these types of equations and develop techniques widely used in the literature. To give a concrete setting in this thesis, we shall use the disease *atherosclerosis* as an example. This is a disease of the arteries which provides a rich area of modelling due to its complex nature. We shall give a review on atherosclerosis in this introduction, look at mathematical modelling of the disease and throughout the thesis discuss the application of the methods we develop in the context of atherosclerosis.

1.1 Circulatory System

The circulatory system is well-studied and some aspects of it are common knowledge. To better approach the subject of atherosclerosis, let us review the relevant features here. There exist many books on the subject (Noble et al. 2010, Tortora & Derrickson 2014, Standring 2008) and the following exposition draws upon them.

The circulatory system consists of the heart, cardiovascular vessels and the lungs. Its main function is to carry nutrients, oxygen and other necessary substances around the body. This is achieved by circulation of blood. Essentially, the heart acts as a pump

that drives the flow of blood around the body and back to the heart again. It is divided into the pulmonary and systemic circulations. In the pulmonary circulation deoxygenated blood is pumped to the lungs where it is oxygenated and then brought back to the heart. The oxygenated blood is then pumped through the systemic circulation to the other parts of the body. Oxygen is given up to cells requiring it, and the deoxygenated blood is transported back to the heart, after which the cycle begins anew.

Blood flows through what are, crudely put, elastic tubes. Blood flowing away from the heart are carried in arteries, whereas blood flowing towards the heart are carried in veins. Arteries close to the heart are thick, and as they become more distant to the heart they branch out and their diameter decreases. At the extremities of the circulation the vessels are extremely small and are called arterioles. These then merge into a network of capillaries where exchange of molecules to the tissues can be achieved via diffusion. The capillaries then transport the blood to small venules which then transport blood back to the heart in vessels that increase in lumen diameter and converge into veins.

The cellular structure of arteries and veins are similar, however important differences exist. Since atherosclerosis is chiefly a disease of the arteries, let us concentrate only on the structure of arteries. Fig. 1.1 shows important features of an artery. The artery is divided into three main layers:

- Intima - made up of endothelium (single layer of endothelial cells) sitting on top of basal lamina which in turn rests upon the subendothelial layer. This is then supported by the internal elastic lamina. The endothelium is coated by the glycocalyx on the luminal side.

- Media - a thick layer containing layers of elastic fibres, collagenous tissue and smooth muscle fibres. Supported by external lamina.

- Adventitia - outermost layer made up of collagenous material, fibroblasts and vasa vasorum

That which surrounds the living cells in the artery is termed the extracellular matrix and is an important part of the tissue which gives the wall structural integrity. It is made up of proteins such as collagen and elastin. These are long proteins that are arranged in fibres. Collagen fibres are resistant to stretching and thus provide structural support of the tissue. They can also attach to the tissue cells thus helping to structure the tissue. Elastin

allows the artery to recoil after it has been stretched. In the media, collagen and elastin fibres are arranged circumferentially to allow the wall to stretch out radially during systole, whereas in the adventitia they are arranged longitudinally to support pressure waves from the pulse. Other constituents of the extra-cellular matrix, such as proteoglycans, attach to cells and other components of the extra-cellular matrix thus allowing organisation, and attract water which provides an environment through which nutrients can diffuse and hydrates cells.

The endothelium is coated by a layer called the glycocalyx. It is a thin layer ($\sim 4\mu$m in the carotid artery) of a mesh rich in proteoglycans and glycoproteins (Reitsma et al. 2007). Glycoproteins come in many varieties, an important class being those which act as receptors in the immune system and those which are known as cell adhesion molecules which help bind cells together to keep structure and to help recruit immune cells in the immune response (see next section). Functions of the endothelium include acting as a permeable barrier to molecules from the blood flow, acting as the site of cell adhesion molecules, regulating coagulation, fibrinolysis and haemostasis (Reitsma et al. 2007, Bernfield et al. 1999).

The endothelium has many important functions. These functions include controlling platelet and leukocyte adhesion, maintaining vascular tone and controlling the passage of molecules through the wall (Davies & Hagen 1993, Pober & Sessa 2007). A major secretion of the endothelial cells is nitric oxide (NO), which has a number of effects including vasodilation and the inhibition of platelet adherence and aggregation, white blood cell (WBC) adhesion and proliferation of vascular smooth muscle cells (SMC's) (Davignon & Ganz 2004). There is evidence that there are components of the glycocalyx layer which generate signals to varying levels of shear stress which then promotes or inhibits endothelial-NO production accordingly (Mochizuki et al. 2002).

Smooth muscle cells are cells that are able to contract and relax in response to various stimuli, including NO. In the blood vessels they help control the lumen diameter. They can also synthesize extra-cellular matrix, and two phenotypes are recognised: the *contractile* phenotype and the *synthetic* phenotype. The synthetic phenotype has a more pronounced presence of rough endoplasmic reticulum which allows it to produce more extra-cellular matrix.

The internal and external elastic lamina are layers of elastic material with fenestrations

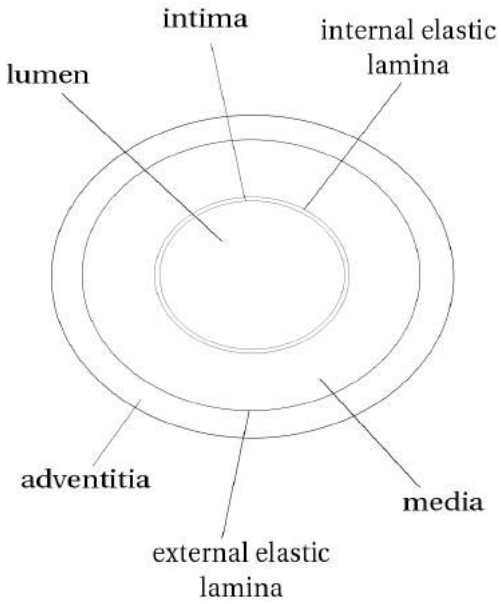


Figure 1.1: Structure of an artery

which allow for the diffusion of molecules between the layers. They may or may not be present in arteries (Stary et al. 1992).

The basal lamina is a thin layer made up of extra-cellular matrix derived from the endothelial cells, a small number of fibroblasts, and some smooth muscle cells. It is rich in non-fibrous proteoglycans (Stary et al. 1992). The subendothelial layer is made up of elastin, collagen and smooth muscle cells (Stary et al. 1992). The smooth muscle cells are mainly of the contractile phenotype. The extra-cellular matrix in the media is derived from the smooth muscle cells and in the adventitia is derived from the fibroblasts.

1.2 Immune Response

Since atherosclerosis is widely regarded as an inflammatory disease, it is instructive to study the immune system. A good introduction to the subject of immunology is provided in Todd & Spickett (2005), where the information from this section comes from.

In order to survive, the body must have some mechanism of dealing with foreign material which can potentially cause damage. For example, organisms such as bacteria can infiltrate the body and release toxic materials, or viruses can kill cells in their replication cycle. It is the role of the immune system to recognise potential threats and eliminate

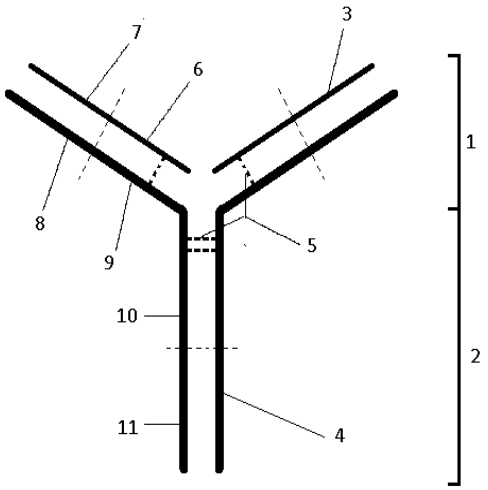


Figure 1.2: Basic structure of an antibody: (1) Fab region; (2) Fc region; (3) light chain; (4) heavy chain; (5) disulphide bonds; (6), (7) light chain constant and variable domains resp.; (8) heavy chain variable domain; (9), (10), (11) heavy chain constant domains.

them.

White blood cells (WBCs), also known as leukocytes, in the blood circulation play a principal role in the immune response. These are cells that recognise and arrange an appropriate response. There are a number of different types of WBC, such as B cells and T cells (which we describe below), phagocytes which ingest foreign material and monocytes which are phagocytes that circulate in the blood that migrate from the blood stream to surrounding tissues, where they differentiate into macrophages and dendritic cells.

Recognition of foreign bodies and infected cells is facilitated by receptor-antigen interactions. Foreign bodies can be identified by certain substances they carry. If a substance can be recognised by a WBC receptor and induce an immune response, the substance is known as an antigen. Receptors on WBC's are protein structures that are expressed on WBC surface membranes. B cell receptors are antibodies (aka immunoglobulins) (see Fig. 1.2). T cell receptors are different but are related through the immunoglobulin superfamily. Receptors have regions where antigens can be bound to the receptor called antigen-binding sites (variable domains in Fig.1.2 - similar variable domains are present in T cell receptors). For each antigen there is a specific binding site - each made different from another by a different sequence of amino acids. B cells recognise antigens in their natural unmodified form, whereas T cells recognise antigens that are processed by

infected cells and subsequently expressed on the surface membrane attached to a HLA (human leukocyte antigen) molecule.

The main function of B cells is to secrete antibodies that identify antigens, immobilise them, and mark them to other WBCs. An immature B-cell (i.e. one that has not been presented with an antigen) expresses only one kind of antibody on its surface. When it encounters an antigen, it is activated, whereupon it matures to a plasma cell and clones itself. Plasma cells secrete soluble forms of antibodies.

When an infected cell presents an antigen, the type of HLA it is bound to is important - HLA class I (found on most cells in the body) interacts with T cells expressing CD8 (CD8+ T cells), and HLA class II (found mainly on other WBCs) interacts with T cells expressing CD4 (CD4+ T cells). T killer (aka cytotoxic) cells are mostly CD8+ and kill infected cells, whereas T helper cells are mostly CD4+ and activate other WBCs. The interaction between HLA and CD4/CD8 is known as signal 1. To become fully activated signal 2 must also be given. This involves interaction between CD28 expressed by the T cell and B7 proteins expressed by the antigen presenting or target cell. Cells presenting antigen to be eventually killed are called target cells, whereas those presenting that eventually activate other WBCs are called antigen presenting cells. Dendritic cells, macrophages and B cells are professional antigen presenting cells that specialise in processing and presenting antigens.

The immune system has various ways to regulate the immune response.

- CTLA-4 expressed by an activated T cell competes more effectively than CD28 to bind with B7 proteins expressed by an antigen presenting cell. In this way T cell activity can be down-regulated.

- WBCs (and other cells such as endothelial cells) secrete proteins known as cytokines which each have a variety of effects. Effects include activation or suppression of other WBCs, regulation of HLA and B7 expression and inducing chemotaxis of other WBCs. Cytokines which produce the latter effect are known as chemokines.

- To aid localisation of immune response at the site of infection, the inflammatory response is set up. This includes vasodilation of blood vessels surrounding infected tissue, increased permeability of vascular walls and chemotaxis of WBCs.

1.3 Cholesterol metabolism

The build-up of lipid in the form of cholesterol esters is central to the progression of atherosclerosis. Clues to how this accumulation occurs may be found by studying the role of lipids and their transport in the body.

Lipids form a class of molecules which includes fats and oils. They are characterised by their resistance to dissolve in water. They are essential to the function of the body, fulfilling functions such as membrane synthesis. Since the body cannot produce all the lipid it requires, some lipid must be obtained through the diet. Once ingested, the only way it can be efficiently transported to parts of the body where it is needed is through the bloodstream. However, due to their insolubilty in water, lipids cannot be transported in their natural form. To overcome this problem, lipids are transported as lipoproteins. These are molecules where the lipid content is bound by a soluble protein coat made up of apoproteins, thus making transport possible. There are many types of lipoproteins, and they are classified according to their relative amount of protein and lipid content. Thus, high density lipoproteins (HDLs) have a larger amount of protein in comparison with lipid, and low density lipoproteins (LDLs) have a larger amount of lipid in comparison with protein.

The lipid most relevant to atherosclerosis is cholesterol, which is the main constituent of plaque. To understand how the cholesterol builds up it is necessary to understand how it is transported in the body. All body cells are capable of synthesising their own cholesterol, but the bulk is obtained through the diet and biosynthesis in the liver. The largest part of cholesterol is transported as LDL in the bloodstream. Hevonoja et al. (2000) describe the structure of LDL in detail: it is made up of esterified cholesterol (40%), apolipoprotein B-100 (20%), free cholesterol (10%) and other lipids. It is a spherical molecule of about 20nm with an outer shell of apo B-100 and free cholesterol encasing the lipids, including esterified cholesterol, in a hydrophobic centre.

But how is cholesterol taken up by cells, and how does the body regulate the levels of cholesterol? In particular, in the progression of atherosclerosis, where does this go wrong? It was these questions which were addressed by Goldstein & Brown (1977) which described the *LDL-pathway* of cholesterol transport. Through a number of in-vitro studies using radioactively labelled LDL, they elucidated the mechanism of cholesterol metabolism via the so-called *LDL receptors*. These are the receptors on cell surface membrane which allow

the cell to 'recognise' LDL molecules and to which they can consequently be bound. This recognition is achieved via a binding site on the LDL receptor which is capable of forming chemical bonds with a specific part of a LDL molecule. Once bound the cell ingests the LDL particle by endocytosis. As the LDL is ingested it is bound by an endosome which is then free to move around the cell cytoplasm. This endosome is taken to a lysosome which contains enzymes which break down the LDL to give amino acids and hydrolyse the esterified cholesterol into free cholesterol.

Thus we have a mechanism for the cellular intake of cholesterol via LDLs. The regulation of cholesterol levels in body cells is controlled by the two enzymes Acyl-CoA cholesterol acyl transferase (ACAT) and 3-hydroxy-3-methylglutaryl coenzyme-A (HMG-CoA) reductase. ACAT promotes the conversion of free cholesterol to less soluble cholesterol esters which form fat droplets in the cytoplasm, and HMG-CoA reductase promotes autoproduction of cholesterol in body cells. As more cholesterol is detected in the cell:

i. ACAT activity is increased so that excess free cholesterol is stored as fat droplets;

ii. less HMG-CoA reductase is produced to avoid excess amounts of cholesterol in cell;

iii. fewer LDL-receptors are produced in endoplasmic reticulum, thus reducing the external intake of cholesterol.

Receptor-mediated endocytosis is not the only way for cholesterol to enter cells: cholesterol may also be ingested into cells by phagocytosis or pinocytosis (unspecific phagosytosis) as well. The implication of this is important in atherosclerosis as it contributes to the accumulation of lipid in the arterial wall. Indeed, immune cells can begin to take up LDL via alternative routes which are not controlled in the way that the LDL-receptor pathway is. We will study the interplay between the immune system and LDL in the arterial wall in more detail in the next chapter.

1.4 The Onset of Atherosclerosis: atherogenesis

There now exists a large body of literature on the hypothesis that atherosclerosis is an inflammatory disease. For reviews on this hypothesis see Ross (1999), Libby et al. (2002), Stoll & Bendszus (2006), Grundtman & Wick (2011).

The concept is as follows: an immune response is set up against LDL derivatives within the intima which then spirals out of control. The intima becomes chronically in-

flamed and a build-up of fatty deposits in the intimal layer occurs. Mechanisms behind this are the focus of this section. Prolonged inflammation leads to the more advanced stages of atherosclerosis, which is the focus of the next section.

The initiation of atherogenesis has been attributed to impaired function of the endothelial layer. There are a number of ways the endothelial can become dysfunctional, and certain risk factors predispose individuals such as hypercholesterolemia, hyperglycemia (associated with diabetes mellitus), hypertension and smoking. In this context, endothelial dysfunction refers to impaired production of nitric oxide (NO). NO is an important regulatory substance which encourages vasodilation, discourages platelet and leukocyte aggregation and affects the permeability of the endothelium (Ross 1999). Reduced levels of NO can derive from the inhibition of eNOS which catalyses the production of NO (Yang & Ming 2006). Reactive oxygen species can rapidly inactivate NO, but on the other hand endothelial dysfunction can also contribute to oxidative stress by creating reactive oxygen species (Cai & Harrison 2000). The disruption of the glycocalyx layer could play a role in endothelial dysfunction since components of the glycocalyx can act as mechanotransducers to wall shear stress and thus help regulate NO synthesis (Mochizuki et al. 2002, Noble et al. 2008, Broekhuizena et al. 2009).

Furthermore, under stressed conditions, endothelial cells have been shown to express heat shock proteins (Mehta et al. 2005). For example, in-vitro experiments have shown expression of HSP-60 in stressed conditions (Grundtman et al. 2011). Damage to the endothelium then ensues by the action of anti-HSP60 antibodies which, due to a high similarity in structure, recognise HSP60 from both foreign and host sources (Foteinos et al. 2005).

The affected permeability of the endothelial wall may then allow substances such as LDL to infiltrate. Once in the intima, LDL can then react with reactive species (Madamanchi et al. 2005) and give rise to oxidised LDL (oxLDL). There are a number of ways in which LDL can be oxidised in vivo and it may undergo progressive stages of oxidation (Matsuura et al. 2008). OxLDL is immunogenic and promotes the recruitment of immune cells (Matsuura et al. 2008). Thus begins inflammation in the intima. As the LDL become more oxidised, they lose their affinity to LDL receptors and instead are recognised by scavenger receptors which are found on macrophages, dendritic cells and

smooth muscle cells (SMCs) (Stephen et al. 2010). There is no auto-regulation of these scavenger receptors as for the LDL receptor described in the previous chapter, so LDL intake is uncontrolled and the cell becomes overladen with cholesterol. In this state the cell is known as a foam cell. Foam cells secrete pro-inflammatory cytokines that attract more monocytes by promoting endothelial cell-adhesion molecule expression.

Shear stress from the blood flow is an important factor in atherogenesis. In response to local conditions, as well as modulation of NO levels, the intima of an artery may thicken. For example, in places of low shear stress (such as bifurcations or high curvature), thickening of the intima reduces the lumen diameter and increases the shear stress (Stary et al. 1992). Intimal thickening in itself does not represent a diseased state, but locations in the arterial tree where intimal thickening occurs most commonly have been found to be more prone to the development of atherosclerosis (Stary et al. 1992, Kolodgie et al. 2007). Intimal thickening is mainly composed of the synthetic phenotype of smooth muscle cell, as opposed to the contractile phenotype. The proteoglycan rich extra-cellular matrix that they produce can trap LDL molecules in the intima and then lead to their subsequent oxidation (Doran et al. 2008).

1.5 The Progression of Atherosclerosis and Clinical Relevance

The initial stage of the disease, i.e. the initial inflammation, is reversible in that the inflammatory response could eventually halt and the lesion regress. However, due to the self-amplifying nature of the disease, inflammation is susceptible to continue and lead to further progression of the disease.

Foam cells have been shown to leave the site of inflammation into the bloodstream in early lesions, but as the disease progresses the relative number of exiting foam cells decreases and they accumulate (Gerrity 1981). The reasons for this decreased efflux of foam cells are not known for certain (Bobryshev 2006). With time, foam cells eventually die and the cholesterol esters they carry are expelled into the extra-cellular environment (Balla et al. 1995, Hegyi et al. 1996). Thus, the lipid core develops. The reasons for foam cell death are not fully understood. Foam cell death can arise as a result of a process known as apoptosis (programmed cell death) which is regulated by complex cell-receptor and cell-cytokine interactions (Stoneman & Bennett 2004), or can be triggered in response

to toxic substances such as ox-LDL (Hegyi et al. 1996). The actual death of foam cells may not actually be as detrimental as the clearing of the apoptotic cells (efferocytosis) (Kockx & Hermanb 2000, Tabas 2010). Inefficient efferocytosis of apoptotic foam cells leads to a buildup of cholesterol esters in the extra-cellular matrix in a distinct focal lipid core.

The lipid core forms partially from foam cell death, but the level of free cholesterol observed is not accounted for by foam cell death. Intracore haemorrhage from vasa vasorum can contribute to the free cholesterol (Virmani et al. 2005) (Michel et al. 2011). Intracore haemorrhage allows red blood cells (RBCs) to infiltrate and supply the core with free cholesterol from their cholesterol rich membranes. In addition to the fatty content of the core, the death of SMCs and ECs by cytokine action or hypoxia, for example, contributes to the plaque volume.

In response to growth factors and cytokines associated with inflammation, SMCs are attracted from the media to the intima above the lipid pool (Newby 2006). The SMCs act as a source of collagen and other extra-cellular matrix components which form a cap over the lipid core. The cap acts as a barrier between the lipid core and the blood flow, preventing further infiltration of monocytes. In this way the inflammation is controlled to some extent.

Initially, the growth of the lesion does not affect the lumen size as the arterial wall dilates to compensate. However, if dilation is impaired (as in endothelial dysfunction) the lesion produces a local stenosis of the artery. This induces irregular flow around the lesion which could further exacerbate the disease.

As well as a local stenosis, atherosclerotic plaque can be the cause of total luminal occlusion. If the fibrous cap fails and ruptures it exposes thrombogenic material in the plaque to the blood flow and blood clotting (thrombosis) ensues. The end result is a blood clot (thrombus). Depending on the severity of the rupture, the thrombus may remain localised on the plaque (mural thrombus) where it can contribute to further stenosis or further weaken the plaque, or it can form well into the lumen. If the thrombus does not produce a local occlusion, it can still travel downstream and occlude an artery further down the bloodstream (thromboembolism).

The picture painted so far is quite bleak. Does the body have any natural defence against atherosclerosis? The answer is yes, and it is the balance between attack and de-

fence which plays a pivitol role in the progression of the disease. Defences include reverse cholesterol transport by HDL - this is a system by which excess cholesterol is taken up by HDL in the blood after which it is taken to the liver to be excreted. A relatively high HDL level is taken to be a sign of good health, though treatments to increase HDL levels do not necessarily result in the regression+ of atherosclerosis. Cyotkines also play an important role. The number of cytokines involved in atherosclerosis is large and their interactions complex (Ait-Oufella et al. 2011). Though some cytokines promote atherosclerosis, others protect against it. These include cytokines which suppress inflammation by deactivating immune cells, and cytokines which inhibit the action of other proinflammatory cytokines.

So far we have discussed many things that can happen during the formation of atherosclerotic plaque. Now we have an overview of the dynamics of the disease, can we delineate the progress of the disease in a way that is clearer? A classification would not only be a convenient framework to use in the study of the disease, but can also be used as a clinical tool to aid diagnosis, prognosis and the determination of suitable treatments.

In a series of reports for the American Heart Association (AHA), Stary et al. laid out their classification of atherosclerotic plaques (Stary et al. 1994, 1995). This is a classification based on a very wide literature study and gives an almost linear progression with each stage designated by a Roman numeral. Their classification distinguishes early (type I-II), intermediate (type III) and advanced lesions (type IV-VI) (see Table 1.1). According to this classification, plaques that are most prone to rupture are of Type V and VI, with most deaths associated with Type VI (Stary et al. 1995).

Later on, Virmani et al. proposed an improvement on this system using descriptive names for each of their classes (Virmani et al. 2000). Their suggestions are based on their study of 241 culprit plaques after sudden coronary death. In this study 52% of deaths were attributed to acute thrombi, and of these approximately 60% were attributed to plaque rupture. Plaque rupture represented the most common cause of death. Their classification distinguishes between “fibrous cap atheroma” and “thin fibrous cap atheroma”, which are essentially of Type V in the Stary classification. The thin fibrous cap atheroma, defined to be one with cap thickness $< 65\mu$m, merited its own distinct class by observations that fibrous plaques that rupture are more often those with cap thinning. Plaque rupture was not the only cause for acute thrombi. Thus, their classification also emphasises erosion

Type	Features
Type I (initial lesion)	intimal thickening with isolated macrophages and lipids (adaptive intimal thickening)
Type II (fatty streak)	intimal thickening with accumulation of lipids in a fatty streak
Type IIa	progression-prone
Type IIb	progression-resistant
Type III (preatheroma)	
Type IV (atheroma)	lipid core forms
Type V	fibrous cap is formed
Type Va (fibroatheroma)	lipid core with fibrous cap
Type Vb (calcific lesion)	lipid core with fibrous cap with calcification
Type Vc (fibrotic lesion)	fibrous cap with lipid core absent
Type VI (complicated lesion)	lesion surface is disrupted with presence of haematoma, haemmorhage or thrombi

Table 1.1: AHA classification of atherosclerosis as laid out in (Stary et al. 1994, 1995)

and calcified nodules as important plaque characteristics related to thrombic occlusion.

The significance of plaque rupture is further supported by a larger study by Redgrave et al. (Redgrave et al. 2006). They derived their results from 526 symptomatic carotid plaques obtained after endarectomy. Again, about 60% of plaques showed signs of rupture. Of these, the most abundant features of the plaque were a large lipid core (78%) , haemmorhage (78%) and increased inflammation, especially cap inflammation (86%). In a later publication, the cap-thickness of 428 of these plaques were investigated (Redgrave et al. 2008). Thinner caps were observed in plaques that had ruptured: the median thickness for ruptured and unruptured plaques was 300μm and 500μm respectively.

Thin fibrous caps are often referred to as "vulnerable" plaques since they are the most vulnerable to rupture. Patients with vulnerable plaques are deemed to be those at most risk. Virmani et al. (Virmani et al. 2006) emphasise that not only should thin fibrous cap atheroma be taken as vulnerable plaque, but all plaques which have the potential for luminal thrombosis. Thus, pathological intimal thickening, (thick) fibrous cap atheroma and calcified plaque, which are liable to erosion, are considered vulnerable (Virmani et al. 2006).

Of course, the plaque must be looked at within the context of the patient. Thus a shift of emphasis from vulnerable plaque to vulnerable patient was made by Naghavi et al. (2003*a*,*b*). They draw attention to the importance of features that do not relate to the structure of the plaque, for example blood content and cardiac muscle function.

In any case, fibrous cap rupture is a significant event in atherosclerosis However, the mechanisms of plaque rupture are still not certain. The studies mentioned above suggest that a plaque with a large lipid core, a thin fibrous cap and active inflammation is more likely to rupture. A number of reasons why may be surmised, such as a thinner fibrous cap is less able to support mechanical stress before failure and inflammation can contribute to instability by degradation of the fibrous cap matrix by various substances such as matrix metalloproteinases (MMPs) derived from inflammatory cells (Finn et al. 2010). However, other markers that can distinguish more dangerous plaques are needed to aid treatment of the disease.

1.6 Atherosclerosis in the clinic

1.6.1 Preventative measures

Preventative measures such as regular exercise, a healthy diet and no smoking remain key to lower predisposition to atherosclerosis and are aimed at lowering hypercholesterolemia, hypertension and irritant substances that exacerbate inflammation. However, there are factors beyond control (e.g. familial hypercholestoremia), but even so, absence of the known risk factors is no guarantee to halt the development of atherosclerosis.

1.6.2 Diagnosis

If a clinician suspects that a patient has atherosclerosis, one of the first tools they use is an ultrasound scan (Armstrong & Bandyk 2010). More specifically, a duplex ultrasound scan combining B-mode (brightness-mode) ultrasound, which provides information about the physical shape and content of the vessel walls, and Doppler ultrasound, which provides information about the flow of the blood (Armstrong & Bandyk 2010). This method of investigation is popular due to its low-cost, portability, safeness and non-invasive nature. A number of important measurements can be made with ultrasound: the degree of stenosis may be measured, most commonly using the NASCET method (NASCET 1991) or the

ECST method (ECST 1998); the intima-media thickness has been shown to be correlated to the risk of developing cardiovascular disease (Burke et al. 1995, Veller et al. 1993, Bots et al. 1997, van den Oord et al. 2013) and may be measured using ultrasound. Other markers that may be obtained for the risk of atherosclerotic disease include the presence of turbulence in the blood flow, peak systolic velocity and end-diastolic velocity (Armstrong & Bandyk 2010, Zierler 2010).

A known issue with ultrasound, however, is its operator-dependent nature (Mikkonen et al. 1996, Alexandrov et al. 1997). Consequently, the use of ultrasound alone may lead to disagreement in the treatment of atherosclerosis (Meadb et al. 2000). When needed, other means of evaluating the artery may be used, such as an X ray computed tomography (CT) scan (Blankensteijn & Kool 2010), magnetic resonance imaging (MRI) (Litt & Carpenter 2010) or an arteriogram (Rana & McLafferty 2010). These provide more accurate representations of the artery, providing higher resolution accuracy of the geometry and more detailed information about the contents of the vessel wall, but expose the patient to radiation, are more expensive and time costly to run and, in the case of an arteriogram, are invasive.

1.6.3 Treatment

Invasive surgical treatments include angioplasty (inflating a balloon in the vessel to "flatten" the plaque and widen the lumen area), stenting (inserting and leaving behind a small structure that leaves the vessel open), endarterectomy (physically removing the plaque) and bypass surgery. In carotid atherosclerosis, carotid endarterectomy is the most often performed invasive surgical procedure. The decision to operate or not is not a trivial one and an important issue is the treatment of asymptomatic carotid plaque (Naylor 2012), i.e. a carotid plaque exhibiting no external symptoms . The NASCET (NASCET 1991) and the ECST (ECST 1998) trials have provided clinicians and surgeons with invaluable criteria to help with the decision. These two large scale randomised trials followed the course of treatment of patients undergoing carotid endarterectomy and those which did not. The overall findings show a significant benefit of performing carotid endarterectomy on patients with high degree stenosis ($> 70\%$ Nascet, $> 80\%$ ECST), each providing a different way to measure the degree of stenosis.

Drug therapy that is used include statins, which inhibit HMG-CoA reductase, and

antiplatelet drugs to prevent thrombus formation. These are aimed at reducing plaque volume by lowering the amount of lipid and prevent thrombus formation.

1.7 Mathematical Models of Atherosclerosis

A complete model of atherosclerosis would include features of inflammation, haemodynamical flow and morphological features of the plaque and the artery wall. Thus, reaction-diffusion, fluid dynamics and soft tissue mechanics are the most relevant tools to use. Can we hope to build a model that incorporates all these where the relevant parameters can be obtained patient by patient? An interesting discussion on this may be found in Taylor & Humphrey (2009). In particular, they discuss the relevant aspects to model, what has been done and how they might be combined in a patient specific way. Before considering further what kind of model we can build in this particular project we should take a look at what models have been proposed so far.

The models proposed in the literature can be subdivided into two broad categories:

Reaction Models - Suppose we have a simple reaction where two reactants A and B react to give a third substance C. Given initial concentrations of the reactants, we are interested in the evolution of the spatial distribution of the mixture through time. A simple relation is given by the law of mass action which states that the rate of reaction is proportional to the concentrations of the reactants. A simple linear approximation to this law is

$$\frac{d[C]}{dt} = k[A][B]\,, \tag{1.1}$$

where $[A]$, $[B]$ and $[C]$ denote the concentrations of A, B and C respectively, and k is known as the rate constant which can be obtained empirically. If no reverse reaction occurs, the law of mass action then gives

$$\frac{d[A]}{dt} = S_A - k[A][B] \quad \frac{d[B]}{dt} = S_B - k[A][B]\,, \tag{1.2}$$

where S_A and S_B denote possible sources (or sinks) of A and B respectively.

Using this basic idea one can build models of atherosclerosis. For example, Cobbold et al. (2002) investigated the attack of LDL and HDL by free radicals in this way. In their model HDL afforded a certain amount of protection from attack to LDL by acting as sacrificial targets, and thus had an atheroprotective role by reducing the amount of

oxLDL produced. In another model, Ougrinovskaia et al. (2010) investigated the interplay between macrophages and modified LDL on lesion growth. Their model suggested that macrophages played a greater role in lesion growth than LDL levels.

If one adds a spatial dimension to the problem, one can model movement of the chemicals through space. In this case, diffusion is often the phenomenon that is important and one then speaks of reaction-diffusion equations. Similar to diffusion, chemotaxis can also be modelled by a similar law. In modelling atherosclerosis the reactants can be substances such as LDL and cytokines or cells such as WBCs and smooth muscle cells. For example, Ibragimov et al. (2005) modelled the inflammatory process of atherogenesis representing the artery as a 2D annulus with fixed boundary and captured features of the focalisation of the lesion and the start of cap formation with chemotaxis. Fok (2011) modelled the growth of a necrotic core by hypoxic macrophage death in a later stage of atherosclerosis. Khatib et al. (2009) modelled atherogenesis as an inflammatory process, and in partcular lesion growth was associated with an instability in their equations. These models. though interesting, still lack important features such as blood flow. Parameters such as monocyte recruitment and oxygen diffusion from the blood flow into the intima were modelled via conditions on the domain boundary. Parameters were either guessed or obtained from in-vitro studies in the literature.

CFD Models - Blood flow is a significant factor in atherosclerosis. It acts as the carrier of crucial substances such as oxygen, LDL and WBCs to the artery wall and, as discussed before, the behaviour of the endothelium is in part determined by the local fluid dynamics on the surface. Thus, fluid dynamics is a key component in modelling arteries.

Arteries form part of the cardiovascular system which consists of the entire arterial tree, venous tree and the heart. In particular, it is closed. However, for reasons of computation and practical interest, one is normally only interested in modelling a few arteries at most. Therefore one is often faced with a geometry of the kind shown in Fig. 1.3. It consists of the lumen domain Ω_f which is occupied by fluid, the solid artery wall domain Γ_w, the fluid inlet and outlet boundaries $\Gamma_{f,in}$ and $\Gamma_{f,out}$, respectively, endothelial wall $\Gamma_{w,end}$, external wall of the artery $\Gamma_{w,ext}$ and the wall cut-off boundaries $\Gamma_{w,bound}$. The boundaries $\Gamma_{f,in}$, $\Gamma_{f,out}$ and $\Gamma_{w,bound}$ are artificial boundaries which result from reducing the model of the whole arterial tree. Care must be taken to model the behaviour of the system at these boundaries to minimize artificial effects.

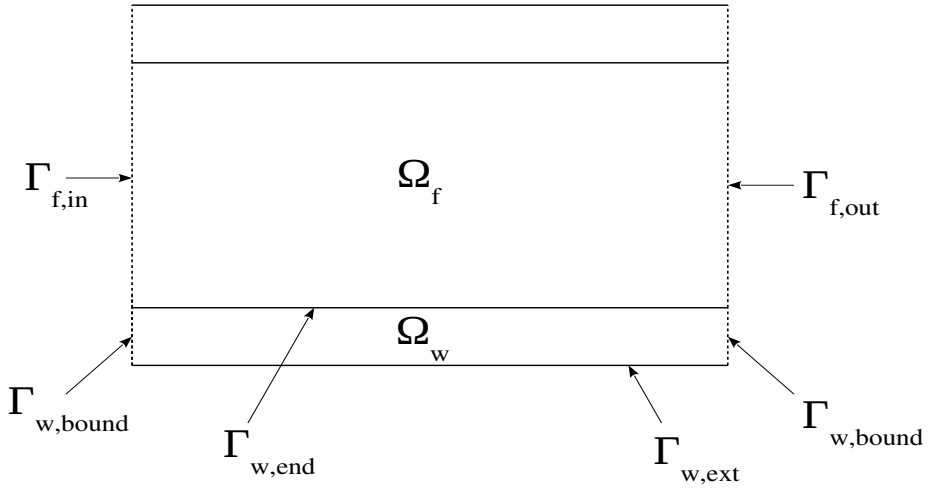


Figure 1.3: Typical 2D computational geometry of an artery segment. For clarity, the top wall has not been labelled.

Let $\mathbf{u}(\mathbf{x})$ be the velocity field of a fluid, where $\boldsymbol{x} \in \Omega$ for some domain Ω. Balance of momentum in a fluid element yields Batchelor (2000):

$$\rho\left(\frac{d\boldsymbol{u}}{dt} + (\boldsymbol{u}\cdot\nabla)\,\boldsymbol{u}\right) = \rho\boldsymbol{F} + \mathbf{div}\boldsymbol{\sigma} \qquad \text{in } \Omega, \tag{1.3}$$

where $\boldsymbol{\sigma}$ is the stress tensor, $\mathbf{F}$ is some external force and ρ is the density of the fluid taken to be constant throughout the domain. In a fluid the stress tensor is dependent on the rate of strain of the fluid, i.e. $\boldsymbol{\sigma} = \boldsymbol{\sigma}\big(\mathbf{D}(\mathbf{u})\big)$ where $\mathbf{D}(\mathbf{u})$ is the rate of strain tensor

$$D_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right). \tag{1.4}$$

A constitutive law is needed to express the dependence of $\boldsymbol{\sigma}$ on $\mathbf{D}(\mathbf{u})$. Here, a number of models for blood can be applied. In the simplest case, blood is treated as a Newtonian fluid, i.e. one in which the components of the stress tensor are linearly proportional to the velocity derivatives. In this case, with incompressibility, the stress tensor has the form

$$\sigma_{ij} = -p\delta_{ij} + 2\mu D_{ij}, \tag{1.5}$$

where p is the pressure and μ is the constant of proportionality known as the viscosity of the fluid. The momentum equation for an incompressible Newtonian fluid then becomes

$$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u}\cdot\nabla)\,\boldsymbol{u} = -\nabla p + \mu\nabla^2\boldsymbol{u} \qquad \text{in } \Omega. \tag{1.6}$$

However, due to the presence of RBCs in the blood plasma, blood is actually a non-Newtonian fluid. In this case a generalised Newtonian model is often used whereby (1.5) holds, but a constitutive law is found for μ which is no longer constant Formaggia et al. (2009).

The Navier-Stokes equations (1.6) must be completed with an intial condition and appropriate boundary conditions on $\partial\Omega$. In cardiovascular applications these are usually Dirichlet and Neumann conditions of the form:

$$\mathbf{u} = \mathbf{g} \qquad \text{on } \Gamma_d \tag{1.7}$$

$$\boldsymbol{\sigma}\mathbf{n} = \mathbf{h} \qquad \text{on } \Gamma_n\,, \tag{1.8}$$

where Γ_d and Γ_n form a partition of Γ and $\mathbf{n}$ is the outward facing unit surface normal.

The above assumes a rigid boundary which does not move in space. However, the kinematics of the artery wall are strongly coupled to that of the blood flow which it surrounds. Whereas the equations of fluid dynamics are most readily presented in the Eulerian frame, it is more natural to consider elastic behaviour of solids in a Lagrangian frame since points in the solid do not move relatively far from their respective positions.

Let $\Omega_0 \subset \mathbb{R}^d$ be some domain occupied at some reference time which, without loss of generality, we can take to be $t = 0$. This is often taken to be the rest configuration which is the configuration of the system in the absence of external forces. A deformation $\boldsymbol{\phi} : \Omega_0 \times T \to \Omega(t)$ displaces each point $\mathbf{x_0} \in \Omega_0$ in the reference configuration after some time t to a point in a new configuration $\mathbf{x}(t) = \boldsymbol{\phi}(\mathbf{x_0}, t) \in \Omega(t)$. The variable of interest here is the displacement $\boldsymbol{\eta} : \Omega_0 \times T \to \mathbb{R}^d$ defined as

$$\boldsymbol{\eta}(\mathbf{x_0}, t) = \boldsymbol{\phi}(\mathbf{x_0}, t) - \mathbf{x_0}. \tag{1.9}$$

It can be shown that $\boldsymbol{\eta}$ satisfies (e.g. see Formaggia et al. (2009))

$$\rho_0 \frac{\partial^2 \boldsymbol{\eta}}{\partial t^2} = \mathbf{div}_{\mathbf{x_0}} \boldsymbol{\Pi} + \rho_0 \mathbf{f_0} \qquad \text{in } \Omega_0\,. \tag{1.10}$$

Here, $\mathbf{div}_{\mathbf{x_0}}$ denotes taking the divergence with derivatives with respect to each spatial coordinate in Ω_0, $\boldsymbol{\Pi} = J\boldsymbol{\sigma}\mathbf{F}^{-T}$ is known as the first Piola-Kirchoff tensor, $\boldsymbol{\sigma}$ is the stress tensor "observed" in the current configuration, $\mathbf{F} = \mathbf{div}_{\mathbf{x_0}}\boldsymbol{\phi}$ is the deformation gradient tensor, $J = \det\mathbf{F}$ and finally $\rho_0 = J\rho(x,t)$ and $\mathbf{f_0} = J\mathbf{f}(x,t)$ where $\rho(x,t)$ and $\mathbf{f}(x,t)$ are the density and applied external force observed in the current configuration respectively.

Similarities may be seen between equation (1.10) and the corresponding equation for fluids (1.3). The role of the stress tensor in (1.3) is played here by the first Piola-Kirchoff tensor $\mathbf{\Pi}$ and in a similar way to fluids, a constitutive relation must be found in order to complete equation (1.10). Such constitutive relations used in modelling biological tissues are discussed in Humphrey (2003) and those specially proposed for modelling arterial tissue are discussed in Holzapfel & Ogden (2010).

With a model for the blood flow one can then model transport behaviour through to the artery wall. Such transport is limited by the relative permeability of the wall to different substances which can be shear dependent. A comprehensive review of the mathematical modelling of transport processes into the intima is given by Tarbell (2003). In the review, a distinction between "fluid-limited" and "wall-limited" transport of species is made: in the former case the fluid dynamics is important in determining the rate of transport through the endothelium, whereas in the latter case the wall properties are more important. This is due to the relative reactivity of the species being transported with the endothelial wall. Thus, a species such as oxygen is taken up readily by the entire endothelial surface and is more likely to be fluid-limited, whereas a species such as LDL does not generally react with the endothelial surface and its transport is more likely to be wall-limited, being determined by the varying permeability of the wall.

The flux of solvent (i.e. blood plasma), J_v, and solute (e.g. LDL or oxygen), J_s, from the lumen through the endothelium is often modelled by the Kedem-Katchalsky equations (Kedem & Katchalsky 1958):

$$J_v = \quad L_p(\nabla P - \sigma \nabla \pi) \tag{1.11}$$

$$J_s = \quad P_0 \nabla C + (1-\sigma) J_v \overline{C}, \tag{1.12}$$

where L_p is the hydraulic conductivity, σ is the reflection coefficient, P_0 is the diffusive permeability, ∇P is the pressure difference between the lumen and endothelium, $\nabla \pi$ is the oncotic pressure difference, ∇C is the solute concentration difference. Using this idea, models have recently been proposed for the early stages of atherosclerosis combining fluid flow and reaction-diffusion equations. Calvez et al. (2010) looked at a 2D model of plaque growth. The blood flow prescribed at the inlet was a steady Poiseuille profile, whilst parameter values were either taken from the literature or guessed. Of note, they captured physical plaque growth by allowing the boundary to grow. Filipovic et al. (2011) looked at a 3D model of inflammation. Sources of parameters were unclear and blood

flow was taken to be steady throughout the entire lumen. Cilla et al. (2014) considered a detailed description of plaque growth, including reaction-convection-diffusion equations to describe the inflammation coupled with blood flow at the interface in an idealised three-dimensional geometry. In their work the phyiscal growth of the plaque is also included and the parameters are derived from experiments in the literature.

In these models blood flow was taken to be steady with a Poiseuille profile at the inlet. This is in direct contrast to actual conditions in vivo where the blood flow is pulsatile. The justification of taking steady flow was that the time-scale of plaque growth (months/years) is much longer than that of pulsatile blood flow (seconds). However, studies have shown that a pulsatile flow and a steady state approximation lead to different transport behaviour especially in stenosed areas of the artery where there is disturbed flow Sun et al. (2007), Liu et al. (2011). As expected, species that are fluid-limited show more marked differences than those that are wall-limited.

1.8 Summary

Although atherosclerosis has been studied extensively, there remain essential questions to be answered. They are made even more important due to the prevalence of the disease today. In particular, the treatment of atherosclerosis is an issue that requires even more research to ensure that every individual receives the most appropriate course of treatment available. This is related to the efficacy of drugs and surgery in each individual. Whilst many trial studies have examined these subjects, there is now a shift of paradigm to provide patient-specific analysis, i.e. to consider each patient individually and base their treatment on their particular characteristics rather than solely on statistical results obtained on others. Thus, mathematical modelling has a role to play in developing models into which patient-specific data can be input and useful information given. These mathematical models will have parameters that need to be calibrated, and in order to be meaningful, in a diagnostic sense, they must be parameters that can be measured accurately and with relative ease. This is one of the biggest challenges in the mathematical modelling of atherosclerosis and every model must therefore consider its parameters with thought.

In the remainder of this thesis, we will lay some groundwork upon which models of atherosclerosis may be built. In particular, we will consider first, reaction-diffusion equations (RDEs) and second, an optimisation problem. We have seen that reaction-

diffusion systems are used to model atherosclerosis, so our findings can be applied to those models. Also, in the 'patient-specific' paradigm of the treatment of atherosclerosis, inverse problems are a natural way to find parameters for models on an individual basis. In Chapter 2 we detail the numerical methods we use to solve the RDEs numerically. Namely, we use the finite element method to discretise in space and a selection of time-stepping schemes to discretise time. We also investigate how to solve the non-linearities in the resulting systems. We will then extend these results to explore a coupled-bulk surface system and perform a linear stability analysis on the system. Finally in Chapter 4 we will consider a shape identification problem which we solve using optimisation.

Chapter 2

Numerical Methods for Reaction-Diffusion Systems

In this chapter, we present robust, efficient and accurate fully implicit time-stepping schemes and nonlinear solvers for numerically solving systems of reaction-diffusion equations. We study fully implicit schemes by use of the Newton method and the Picard iteration applied to the backward Euler, the Crank-Nicolson (and its modifications) and the fractional-step θ methods. We conclude that the fractional-step θ method coupled with a single Newton iteration at each timestep is as accurate as the fully adaptive Newton method; and both outperform the Picard iteration. In particular, the results strongly support the observation that a single Newton iteration is sufficient to yield as accurate results as those obtained by use of an adaptive Newton method. This is particularly advantageous when solving highly complex nonlinear partial differential equations on evolving domains and surfaces. To validate our theoretical results, various appropriate numerical experiments are exhibited on stationary planary domains and in the bulk of stationary surfaces.

One revealing result of our studies is that, for RDEs, a single Newton iteration is sufficient to obtain as accurate solutions as when an adaptive Newton scheme is used. This single Newton iteration for the nonlinear reaction kinetics outperforms a single Picard iteration applied to IMEX time-stepping schemes used by Madzvamuse (2006). On the other hand, the overall elapsed CPU time taken by the Picard iteration is substantially lower than that taken by the Newton method for the backward Euler and Crank-Nicholson and its modifications. The only exception is the fractional θ-method where the Newton

method outperforms the Picard iteration in terms of the overall elapsed CPU time. This is attributed to the fact that the Newton method uses the GMRES solver to invert the resulting non-symmetric matrices, whereas the Picard iteration uses the CG solver to invert block diagonal symmetric matrices. Our recommendation is that when solving RDEs on stationary and evolving domains and surfaces using fully implicit schemes, it is sufficient to employ a single Newton iteration with the fractional θ-method in order to obtain as accurate solutions as those obtained when an adaptive Newton method is used. This is particularly relevant for problems posed on evolving domains and surfaces (Elliott & Ranner 2012, Lakkis et al. 2013). Numerical experiments demonstrate that the fractional θ-method coupled with a single Newton method is about 130 times faster than the backward Euler and 15 times faster than the Crank-Nicholson and its modifications.

Hence, this chapter is structured as follows: in Section 2.1 we present the model equations under study and the choice of parameter values to be used. The space and time discretisation methods and schemes are detailed in Section 2.2. To validate our theoretical predictions, we present experimental order of convergence results in Section 2.4. Various numerical experiments are presented and discussed in Section 2.5. In Section 2.6 we demonstrate how fully implicit schemes and appropriate nonlinear solvers outperform standard IMEX schemes. To demonstrate the applicability and generality of the fully implicit scheme, in Section 2.7 we present various solutions of RDEs on stationary domains and in the bulk of stationary surfaces. Finally, we conclude and outline future research in Section 2.8

2.1 The model equations

Let Ω be a convex domain with Lipschitz boundary, $I = [0, T]$ be some time interval and $(\boldsymbol{x}, t) \in \Omega \times I$. We take for illustrative purposes the well-known Schnakenberg reaction kinetics (also known as the *activator-depleted* substrate or Brusselator model (Gierer & Meinhardt 1972, Prigogine & Lefever 1968*a*, Schnakenberg 1979)) to obtain the model system of RDEs which reads

$$\begin{cases} \dfrac{\partial u}{\partial t} - \nabla^2 u = \gamma(a - u + u^2 v) := \gamma f(u, v), \\ \dfrac{\partial v}{\partial t} - d\nabla^2 v = \gamma(b - u^2 v) := \gamma g(u, v), \qquad \text{in } \Omega \times I, \end{cases} \tag{2.1.1}$$

for the pair $(u(\boldsymbol{x}, t), v(\boldsymbol{x}, t))$ and some real positive numbers a, b, d and γ. Here, u and v correspond to concentrations of two chemical species, d measures the ratio of the relative

diffusivity of the v to u and γ measures the strength of the reaction. To close the system, we choose homogeneous Neumann boundary conditions on the entire boundary and take initial conditions to be small random perturbations around the steady state values

$$(u_{eq}\,, v_{eq}) = \left(a + b\,,\ \frac{b}{(a+b)^2}\right). \tag{2.1.2}$$

Next, we recollect briefly conditions for diffusion-driven instability (Turing 1952) used to guide parameter estimation as well as yielding linear approximate solutions close to bifurcation points which validate numerical results. With no spatial variation in u or v, in the absence of diffusion, the equilibrium point (2.1.2) is linearly stable provided that (Murray 2003)

$$f_u + g_v < 0 \quad \text{and} \quad f_u g_v - f_v g_u > 0, \tag{2.1.3}$$

where the derivatives are evaluated at the equilibrium point. If one then allows spatial inhomogeneity, it is possible that the system evolves to an inhomogeneous steady state. This entails an initial instability caused by diffusion, a phenomenon known as diffusion-driven instability, or Turing instability after the author who first described it (Turing 1952). Subsequently, the non-linear reaction terms keep the solution bounded in an invariant set (Smoller 1994).

A linear stability analysis reveals conditions that will drive a Turing instability (Murray 2003). Let us consider small perturbations from the equilibrium and write them as $\tilde{u}(\boldsymbol{x}, t) = u(\boldsymbol{x}, t) - u_{eq}$ and $\tilde{v}(\boldsymbol{x}, t) = v(\boldsymbol{x}, t) - v_{eq}$. Define $\boldsymbol{\xi}(\boldsymbol{x}, t) = (\tilde{u}(\boldsymbol{x}, t), \tilde{v}(\boldsymbol{x}, t))$ and

$$\boldsymbol{A} = \begin{pmatrix} f_u & f_v \\ g_u & g_v \end{pmatrix} \quad \text{and} \quad \boldsymbol{D} = \begin{pmatrix} 1 & 0 \\ 0 & d \end{pmatrix}, \tag{2.1.4}$$

where the derivatives are evaluated at the equilibrium values. Then linearising (2.1.1) we obtain the system of linear PDEs

$$\frac{\partial \boldsymbol{\xi}(\boldsymbol{x}, t)}{\partial t} = \gamma \boldsymbol{A} \boldsymbol{\xi}(\boldsymbol{x}, t) + D \nabla^2 \boldsymbol{\xi}(\boldsymbol{x}, t), \tag{2.1.5}$$

with homogeneous Neumann boundary conditions (Murray 2003). This system can be solved by separation of variables to yield

$$\boldsymbol{\xi}(\boldsymbol{x}, t) = \sum_k \boldsymbol{c_k} e^{\lambda t} \xi_k(\boldsymbol{x}), \tag{2.1.6}$$

where $\boldsymbol{c_k}$ are some constant vectors and $\xi_k(\boldsymbol{x})$ are the modes which solve the homogeneous Neumann problem

$$\nabla^2 \xi_k(\boldsymbol{x}) + k^2 \xi_k(\boldsymbol{x}) = 0. \tag{2.1.7}$$

Thus, from (2.1.5) we have

$$\lambda \boldsymbol{I} = \gamma \boldsymbol{A} + k^2 \boldsymbol{D}, \tag{2.1.8}$$

where $\boldsymbol{I}$ is the identity matrix. This system has non-trivial solutions if and only if

$$| \lambda \boldsymbol{I} - \gamma \boldsymbol{A} - k^2 \boldsymbol{D} | = 0. \tag{2.1.9}$$

These modes will decay with time unless their wavenumber k lies in the range

$$k_-^2 < k^2 < k_+^2, \tag{2.1.10}$$

where

$$k_\pm = \gamma \frac{df_u + g_v \pm \sqrt{(df_u + g_v)^2 - 4d(f_u g_v - f_v g_u)}}{2d}, \tag{2.1.11}$$

and the following conditions hold:

$$df_u + g_v > 0 \quad \text{and} \quad (df_u + g_v)^2 - 4d(f_u g_v - f_v g_u) > 0, \tag{2.1.12}$$

where the derivatives are evaluated at the equilibrium values. Thus, if we perturb the system from equilibrium, under certain choices of parameters, we can expect exponential growth of some modes which correspond to the linearly unstable modes of (2.1.6). This restriction of parameters defines the so-called Turing space (Turing 1952, Murray 2003).

When Ω is the two-dimensional unit square, the eigenmodes of (2.1.7) have the form $\cos(n\pi x)\cos(m\pi y)$ for $n, m \in \mathbb{Z}$ with eigenvalues $k^2 = n^2 + m^2$. The choices $a = 0.1$, $b = 0.9$, $d = 10$ and $\gamma = 29$ will lead to diffusion-driven instability (Madzvamuse 2000). The parameters chosen guarantee that the modes corresponding to $n^2 + m^2 = 1$ are linearly unstable. Furthermore, it can be calculated that the corresponding exponential growth rate is about 1.6246 (Murray 2003). This exponential growth rate will later serve as an aid to see if our numerical simulations are consistent with the theory.

2.2 Numerical methods

Due to the non-linearites, an analytical solution to (2.1.1) is not readily available, and so we seek numerical solutions using the finite element method as detailed below.

2.2.1 Weak Formulation

Let $\varphi \in H^1(\Omega)$. Then the weak form is as follows: find solutions $u, v \in L^2(0,t;H^1(\Omega))$ such that for every test function φ we have

$$\begin{cases} \displaystyle\int_\Omega (u_t\varphi + \nabla u \cdot \nabla\varphi) = \gamma \int_\Omega (a - u + u^2 v)\varphi, \\ \\ \displaystyle\int_\Omega (v_t\varphi + d\nabla v \cdot \nabla\varphi) = \gamma \int_\Omega (b - u^2 v)\varphi \end{cases} \qquad \forall \boldsymbol{x} \text{ on } \Omega,\ t > 0, \cdot \tag{2.2.1}$$

In the above, we have used Green's identities with zero Neumann boundary conditions.

2.2.2 Finite element spatial discretisation

To discretise in space we use the finite element method (FEM) (Hughes 2003, Thomée 2006). We shall experiment with a few time discretisation methods. Time-stepping schemes for this reaction system were investigated by Madzvamuse (2006) for both stationary and evolving domains. The treatment of the non-linear terms was semi-implicit - here we shall treat them implicitly, thereby granting greater stability.

2.2.3 Space discretisation

Let T_h be a triangulation of $\Omega_h \subseteq \Omega$, and let each node have coordinates $\boldsymbol{x}_i$, $i = 1, 2, \cdots, N_h$, with N_h denoting the number of degrees of freedom. Le us take the finite dimensional subspace $V_{\Omega_h} \subset H^1(\Omega_h)$. Then we seek $u_h, v_h \in L^2(0,t;V_{\Omega_h})$ such that for all $\varphi_h \in V_{\Omega_h}$

$$\begin{cases} \displaystyle\int_{\Omega_h} (u_{h,t}\varphi_h + \nabla u_h \cdot \nabla\varphi_h) = \gamma \int_{\Omega_h} (a - u_h + u_h^2 v_h)\varphi_h, \\ \\ \displaystyle\int_{\Omega_h} (v_{h,t}\varphi_h + d\nabla v_h \cdot \nabla\varphi_h) = \gamma \int_{\Omega_h} (b - u_h^2 v_h)\varphi_h, \end{cases} \qquad \forall \boldsymbol{x} \text{ on } \Omega_h,\ t > 0,$$

Let $\{\varphi_i\}_{i=1}^{N_h}$ be the set of piecewise linear shape functions on Ω_h. Then the set forms a basis of V_Ω. Thus, we can write the discrete solution variables as $u_h = \Sigma_{i=1}^{N_h} u_i\varphi_i := \boldsymbol{u}\cdot\boldsymbol{\varphi}$, and similarly for v_h. Then, the above can be written compactly in matrix form as

$$\begin{cases} M\dot{\boldsymbol{u}} + \gamma M\boldsymbol{u} + A\boldsymbol{u} - \gamma B(\boldsymbol{u},\boldsymbol{v})\boldsymbol{u} = \gamma a \mathbb{1}_\phi, \\ \\ M\dot{\boldsymbol{v}} + dA\boldsymbol{v} + \gamma B(\boldsymbol{u},\boldsymbol{u})\boldsymbol{v} = \gamma b \mathbb{1}_\phi, \end{cases} \tag{2.2.2}$$

where A and M are the stiffness and mass matrices respectively with entries

$$a_{ij} = \int_\Omega \nabla\phi_i \cdot \nabla\phi_j \, d\boldsymbol{x} \quad \text{and} \quad m_{ij} = \int_\Omega \phi_i\phi_j \, d\boldsymbol{x}, \tag{2.2.3}$$

$\mathbb{1}_\phi$ is the column vector with j-th entry ϕ_j. Given some vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, $B(\boldsymbol{a},\boldsymbol{b})$ is the matrix with entries

$$B_{ij} = \sum_{k=1}^{N_h}\sum_{l=1}^{N_h} a_k b_l \int_\Omega \phi_k\phi_l\phi_i\phi_j \, d\boldsymbol{x}. \tag{2.2.4}$$

It is readily checked that given a third vector $\boldsymbol{c}$, the matrix B satisfies

$$B(\boldsymbol{a},\boldsymbol{b})\boldsymbol{c} = B(\boldsymbol{a},\boldsymbol{c})\boldsymbol{b} = B(\boldsymbol{c},\boldsymbol{b})\boldsymbol{a}. \tag{2.2.5}$$

Equation (2.2.2) does not yet lend itself to a numerical solution. First, it is still continuous with respect to time. Second, the non-linear matrix B does not allow a solution to be gained in a straightforward manner.

2.2.4 Time discretisation

For illustrative purposes, we will consider the backward-Euler (BE) and Crank-Nicolson (CN) methods and the fractional-step θ-scheme (FSTS) (Glowinski 2003) with a uniform timestep τ. To proceed, define

$$\boldsymbol{G}_1(\boldsymbol{u},\boldsymbol{v}) = A\boldsymbol{u} + \gamma M\boldsymbol{u} - \gamma B(\boldsymbol{u},\boldsymbol{v})\boldsymbol{u} - \gamma a\mathbb{1}_\phi \tag{2.2.6}$$

$$\text{and} \quad \boldsymbol{G}_2(\boldsymbol{u},\boldsymbol{v}) = dA\boldsymbol{v} + \gamma B(\boldsymbol{u},\boldsymbol{u})\boldsymbol{v} - \gamma b\mathbb{1}_\phi\,. \tag{2.2.7}$$

For the BE discretisation we solve at the $(n+1)$-th timestep

$$\begin{cases} M\dfrac{\boldsymbol{u}^{n+1}-\boldsymbol{u}^n}{\tau} + \boldsymbol{G}_1(\boldsymbol{u}^{n+1},\boldsymbol{v}^{n+1}) = \boldsymbol{0}, \\ M\dfrac{\boldsymbol{v}^{n+1}-\boldsymbol{v}^n}{\tau} + \boldsymbol{G}_2(\boldsymbol{u}^{n+1},\boldsymbol{v}^{n+1}) = \boldsymbol{0}. \end{cases} \tag{2.2.8}$$

For the CN we solve

$$\begin{cases} M\dfrac{\boldsymbol{u}^{n+1}-\boldsymbol{u}^n}{\tau} + \dfrac{1}{2}\left[\boldsymbol{G}_1(\boldsymbol{u}^{n+1},\boldsymbol{v}^{n+1}) + \boldsymbol{G}_1(\boldsymbol{u}^{n},\boldsymbol{v}^{n})\right] = \boldsymbol{0}, \\ M\dfrac{\boldsymbol{v}^{n+1}-\boldsymbol{v}^n}{\tau} + \dfrac{1}{2}\left[\boldsymbol{G}_2(\boldsymbol{u}^{n+1},\boldsymbol{v}^{n+1}) + \boldsymbol{G}_2(\boldsymbol{u}^{n},\boldsymbol{v}^{n})\right] = \boldsymbol{0}. \end{cases} \tag{2.2.9}$$

For the FSTS, we split the differential operators, i.e. $\boldsymbol{G_1}$ and $\boldsymbol{G_2}$, into two parts. In this problem, there is the natural dichotomy of diffusion and reaction; however, here, we shall simply make the distinction between terms that are linear and those that are not. Once we have this separation, we split the timestep into three unequal portions alternately treating each dividend implicitly and explicitly. For some $\theta \in (0, 1/2)$, we first solve for the pair $(\boldsymbol{u}^{n+\theta}, \boldsymbol{v}^{n+\theta})$

$$\begin{cases} M\dfrac{\boldsymbol{u}^{n+\theta}-\boldsymbol{u}^n}{\theta\tau} + A\boldsymbol{u}^{n+\theta} + \gamma M\boldsymbol{u}^{n+\theta} = \gamma a\mathbb{1}_\phi + \gamma B(\boldsymbol{u}^n,\boldsymbol{v}^n)\boldsymbol{u}^n, \\ M\dfrac{\boldsymbol{v}^{n+\theta}-\boldsymbol{v}^n}{\theta\tau} + dA\boldsymbol{v}^{n+\theta} = \gamma b\mathbb{1}_\phi - \gamma B(\boldsymbol{u}^n,\boldsymbol{u}^n)\boldsymbol{v}^n, \end{cases} \tag{2.2.10}$$

then for the pair $(\boldsymbol{u}^{n+1-\theta}, \boldsymbol{v}^{n+1-\theta})$ we solve

$$\begin{cases} M\dfrac{\boldsymbol{u}^{n+1-\theta}-\boldsymbol{u}^{n+\theta}}{(1-2\theta)\tau} - \gamma B(\boldsymbol{u}^{n+1-\theta}, \boldsymbol{v}^{n+1-\theta})\boldsymbol{u}^{n+1-\theta} = \gamma a \mathbb{1}_\phi - A\boldsymbol{u}^{n+\theta} - \gamma M \boldsymbol{u}^{n+\theta}, \\ M\dfrac{\boldsymbol{v}^{n+1-\theta}-\boldsymbol{v}^{n+\theta}}{(1-2\theta)\tau} + \gamma B(\boldsymbol{u}^{n+1-\theta}, \boldsymbol{u}^{n+1-\theta})\boldsymbol{v}^{n+1-\theta} = \gamma b \mathbb{1}_\phi - dA\boldsymbol{v}^{n+\theta}, \end{cases} \tag{2.2.11}$$

and finally for the pair $(\boldsymbol{u}^{n+1}, \boldsymbol{v}^{n+1})$ we solve

$$\begin{cases} M\dfrac{\boldsymbol{u}^{n+1}-\boldsymbol{u}^{n+1-\theta}}{\theta\tau} + A\boldsymbol{u}^{n+1} + \gamma M\boldsymbol{u}^{n+1} = \gamma a \mathbb{1}_\phi + \gamma B(\boldsymbol{u}^{n+1-\theta}, \boldsymbol{v}^{n+1-\theta})\boldsymbol{u}^{n+1-\theta}, \\ M\dfrac{\boldsymbol{v}^{n+1}-\boldsymbol{v}^{n+1-\theta}}{\theta\tau} + dA\boldsymbol{v}^{n+1} = \gamma b \mathbb{1}_\phi - \gamma B(\boldsymbol{u}^{n+1-\theta}, \boldsymbol{u}^{n+1-\theta})\boldsymbol{v}^{n+1-\theta}. \end{cases} \tag{2.2.12}$$

Note that the terms treated implicitly are collected on the LHS and those treated explicitly are collected on the RHS in (2.2.10-2.2.12). In the first and third step, the non-linear terms are treated explicitly and the linear terms implicitly. Conversely, in the second step, the non-linear terms are treated implicitly and the linear terms explicitly. We take $\theta = 1 - 1/\sqrt{2}$ since it can be shown that this method is second order convergent in time for this value of θ - for example, see (Glowinski 2003).

2.2.5 Techniques for treating the non-linearities

To deal with the non-linearities, we shall use the Picard iteration and the Newton method - see, for example (Quarteroni et al. 2006).

At the $(n+1)$-th timestep, the $(k+1)$-th Picard iterate of the BE system (2.2.8) is the solution of the matrix system

$$\begin{pmatrix} \left(\frac{1}{\tau}+\gamma\right)M + A - \gamma B(\boldsymbol{u}_k^{n+1}, \boldsymbol{v}_k^{n+1}) & \mathbf{0} \\ \mathbf{0} & \frac{1}{\tau}M + dA + \gamma B(\boldsymbol{u}_k^{n+1}, \boldsymbol{u}_k^{n+1}) \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_{k+1}^{n+1} \\ \boldsymbol{v}_{k+1}^{n+1} \end{pmatrix} = \begin{pmatrix} \gamma a \mathbb{1}_\phi + \frac{1}{\tau}M\boldsymbol{u}^n \\ \gamma b \mathbb{1}_\phi + \frac{1}{\tau}M\boldsymbol{v}^n \end{pmatrix}. \tag{2.2.13}$$

We note here that the IMEX scheme considered in Madzvamuse (2006) is equivalent to taking a single Picard iteration per timestep. Now, take the left-hand side (LHS) of (2.2.8)

and define

$$\boldsymbol{F}_1(\boldsymbol{u}^{n+1},\boldsymbol{v}^{n+1}) = \left[\left(\frac{1}{\tau}+\gamma\right)M + A - \gamma B(\boldsymbol{u}^{n+1},\boldsymbol{v}^{n+1})\right]\boldsymbol{u}^{n+1} - \gamma a\mathbb{1}_\phi - \frac{1}{\tau}M\boldsymbol{u}^n, \tag{2.2.14}$$

$$\boldsymbol{F}_2(\boldsymbol{u}^{n+1},\boldsymbol{v}^{n+1}) = \left[\frac{1}{\tau}M + dA + \gamma B(\boldsymbol{u}^{n+1},\boldsymbol{u}^{n+1})\right]\boldsymbol{v}^{n+1} - \gamma b\mathbb{1}_\phi - \frac{1}{\tau}M\boldsymbol{v}^n, \tag{2.2.15}$$

and let $\boldsymbol{F}=(\boldsymbol{F}_1,\boldsymbol{F}_2)$. Then we wish to solve $\boldsymbol{F}(\boldsymbol{u}^{n+1},\boldsymbol{v}^{n+1})=\boldsymbol{0}$. The $(k+1)$-th Newton iterate of the BE system is the solution of

$$J_{\boldsymbol{F}}|_{(\boldsymbol{u}_k^{n+1},\boldsymbol{v}_k^{n+1})}\left(\boldsymbol{u}_{k+1}^{n+1}-\boldsymbol{u}_k^{n+1},\boldsymbol{v}_{k+1}^{n+1}-\boldsymbol{v}_k^{n+1}\right) = -\boldsymbol{F}(\boldsymbol{u}_k^{n+1},\boldsymbol{v}_k^{n+1}),\quad i=1,2, \tag{2.2.16}$$

where $J_{\boldsymbol{F}}$ is the Jacobian matrix of $\boldsymbol{F}$ and the column vector $\left(\boldsymbol{u}_{k+1}^{n+1}-\boldsymbol{u}_k^{n+1},\boldsymbol{v}_{k+1}^{n+1}-\boldsymbol{v}_k^{n+1}\right)$ is understood to be the vertical concatenation of the first and second arguements. Using property (2.2.5) of the matrix B, we have for some vector $\boldsymbol{\xi}\in\mathbb{R}^{N_h}$ the Gâteaux derivative

$$\frac{\partial \boldsymbol{F}_1(\boldsymbol{u}_k^{n+1},\boldsymbol{v}_k^{n+1})}{\partial \boldsymbol{u}^{n+1}}\boldsymbol{\xi} = \left[\left(\frac{1}{\tau}+\gamma\right)M + A - 2\gamma B(\boldsymbol{u}_k^{n+1},\boldsymbol{v}_k^{n+1})\right]\boldsymbol{\xi}. \tag{2.2.17}$$

Similarly we have,

$$\frac{\partial \boldsymbol{F}_1(\boldsymbol{u}_k^{n+1},\boldsymbol{v}_k^{n+1})}{\partial \boldsymbol{v}^{n+1}}\boldsymbol{\xi} = -\gamma B(\boldsymbol{u}_k^{n+1},\boldsymbol{u}_k^{n+1})\boldsymbol{\xi}, \tag{2.2.18}$$

$$\frac{\partial \boldsymbol{F}_2(\boldsymbol{u}_k^{n+1},\boldsymbol{v}_k^{n+1})}{\partial \boldsymbol{u}^{n+1}}\boldsymbol{\xi} = 2\gamma B(\boldsymbol{u}_k^{n+1},\boldsymbol{v}_k^{n+1})\boldsymbol{\xi}, \tag{2.2.19}$$

$$\text{and}\quad \frac{\partial \boldsymbol{F}_2(\boldsymbol{u}_k^{n+1},\boldsymbol{v}_k^{n+1})}{\partial \boldsymbol{v}^{n+1}}\boldsymbol{\xi} = \left[\frac{1}{\tau}M + dA + \gamma B(\boldsymbol{u}_k^{n+1},\boldsymbol{u}_k^{n+1})\right]\boldsymbol{\xi}. \tag{2.2.20}$$

Thus, at the $(n+1)$-th timestep, the $(k+1)$-th Newton iterate (2.2.16) is the solution of the system

$$\begin{pmatrix} \left(\frac{1}{\tau}+\gamma\right)M + A - 2\gamma B(\boldsymbol{u}_k^{n+1},\boldsymbol{v}_k^{n+1}) & -\gamma B(\boldsymbol{u}_k^{n+1},\boldsymbol{u}_k^{n+1}) \\ 2\gamma B(\boldsymbol{u}_k^{n+1},\boldsymbol{v}_k^{n+1}) & \frac{1}{\tau}M + dA + \gamma B(\boldsymbol{u}_k^{n+1},\boldsymbol{u}_k^{n+1}) \end{pmatrix}\begin{pmatrix} \boldsymbol{u}_{k+1}^{n+1} \\ \boldsymbol{v}_{k+1}^{n+1} \end{pmatrix}$$

$$= \begin{pmatrix} \gamma a\mathbb{1}_\phi + \frac{1}{\tau}M\boldsymbol{u}^n - 2\gamma B(\boldsymbol{u}_k^{n+1},\boldsymbol{v}_k^{n+1})\boldsymbol{u}_k^{n+1} \\ \gamma b\mathbb{1}_\phi + \frac{1}{\tau}M\boldsymbol{v}^n + 2\gamma B(\boldsymbol{u}_k^{n+1},\boldsymbol{u}_k^{n+1})\boldsymbol{v}_k^{n+1} \end{pmatrix}. \tag{2.2.21}$$

We can easily show that the Jacobian is not singular in our case. Indeed, suppose that $J_{\boldsymbol{F}}|_{(\boldsymbol{u}_k^{n+1},\boldsymbol{v}_k^{n+1})}\boldsymbol{\eta}=\boldsymbol{0}$ for some vector $\boldsymbol{\eta}\in\mathbb{R}^{2N_h}$. Writing $\boldsymbol{\eta}=(\boldsymbol{\eta}_1,\boldsymbol{\eta}_2)$, with $\boldsymbol{\eta}_1$, $\boldsymbol{\eta}_2\in\mathbb{R}^{N_h}$, we

can add the two resulting equations and rearrange to obtain

$$\begin{aligned}\boldsymbol{\eta}_1 &= -\left[\left(\frac{1}{\tau}+\gamma\right)M + A\right]^{-1}\left(\frac{1}{\tau}M + dA\right)\boldsymbol{\eta}_2 \\ &:= -K_1^{-1}K_2\,\boldsymbol{\eta}_2,\end{aligned} \tag{2.2.22}$$

where we have used the fact that A and M are both positive definite matrices and are, therefore, invertible. Upon substitution, this then implies that

$$K_2^{-1}\Big(2\gamma B(\boldsymbol{u}_k^{n+1}, \boldsymbol{v}_k^{n+1})K_1^{-1}K_2 - \gamma B(\boldsymbol{u}_k^{n+1}, \boldsymbol{u}_k^{n+1})\Big)\boldsymbol{\eta}_2 = \boldsymbol{\eta}_2. \tag{2.2.23}$$

If $\boldsymbol{\eta}_2 \neq \mathbf{0}$, then the matrix on the LHS of (2.2.23) must be the identity matrix. However, this cannot be since the coefficients $\boldsymbol{u}_k^{n+1}$ and $\boldsymbol{v}_k^{n+1}$ appear on the LHS but, obviously, do not appear in the identity matrix. Thus, $\boldsymbol{\eta}_2$ must be the zero vector. Similarly, $\boldsymbol{\eta}_1$ must also be the zero vector, from which we conclude that $J_{\boldsymbol{F}}$ is invertible. A similar argument will show that the relevant matrix for Picard iteration is also invertible, though it does not indicate whether or not the iterations converge to the desired solution. For this we need to show that we have a contraction mapping. Indeed, suppose we have a system of PDEs of the form

$$\begin{cases} \dfrac{\partial u}{\partial t} - d_1\nabla^2 u = c_1 + \gamma f(u,v)u, \\ \dfrac{\partial v}{\partial t} - d_2\nabla^2 v = c_2 + \gamma g(u,v)v, \qquad \text{in } \Omega\,, \text{for } t > 0\,, \end{cases} \tag{2.2.24}$$

for the diffusion coefficients $d_1,\ d_2 > 0$, constant source terms $c_1,\ c_2 \in \mathbb{R}$ and functions $f : \Omega \to \mathbb{R}$ and $g : \Omega \to \mathbb{R}$ which are in general non-linear, but globally Lipshitz continuous with Lipshitz constants L_f and L_g respectively, and with no explicit dependence on x or t. Even though the Schnakenberg non-linearities may not be globally Lipshitz in general, the solution remains in an invariant set and therefore a Lipschitz constant may still be found in this region (Venkataraman et al. 2011). We further suppose that u and v always remain positive and that f and g are bounded above by some constants β_f and β_g respectively. Let T_h be a quasi-uniform triangulation of Ω, and S_h be the space of continuous functions on T_h that are piecewise linear on each element. Recasting into the weak form and using the BE method to discretise time as above we solve for $(u^{n+1}, v^{n+1}) \in S_h \times S_h$ from the system

$$\begin{cases} (\dfrac{u^{n+1}-u^n}{\tau}, \phi_h) + d_1(\nabla u^{n+1}, \nabla\phi_h) = (c_1, \phi_h) + (f(u^{n+1}, v^{n+1})u^{n+1}, \phi_h), \\ (\dfrac{v^{n+1}-v^n}{\tau}, \phi_h) + d_2(\nabla v^{n+1}, \nabla\phi_h) = (c_2, \phi_h) + (g(u^{n+1}, v^{n+1})v^{n+1}, \phi_h), \quad \forall\phi_h \in S_h, \end{cases} \tag{2.2.25}$$

where $(\cdot,\cdot)$ denotes the inner product in L_2. We would like to use Picard iteration to solve the non-linear equation iteratively, each iteration requiring a linear problem to be solved. The $(k+1)$-th iterate is then

$$\begin{cases} \left(\dfrac{u_{k+1}^{n+1}-u^n}{\tau},\phi_h\right)+d_1(\nabla u_{k+1}^{n+1},\nabla\phi_h)=(c_1,\phi_h)+\left(f(u_k^{n+1},v_k^{n+1})u_{k+1}^{n+1},\phi_h\right), \\ \left(\dfrac{v_{k+1}^{n+1}-v^n}{\tau},\phi_h\right)+d_2(\nabla v_{k+1}^{n+1},\nabla\phi_h)=(c_2,\phi_h)+\left(g(u_k^{n+1},v_k^{n+1})v_{k+1}^{n+1},\phi_h\right). \end{cases} \tag{2.2.26}$$

Subtracting (2.2.25) from (2.2.26) and choosing appropriate ϕ_h we then have

$$\begin{cases} \dfrac{1}{\tau}||u_{k+1}-u^{n+1}||^2\le\left(f(u_k,v_k)u_{k+1}-f(u^{n+1},v^{n+1})u^{n+1},u_{k+1}-u^{n+1}\right), \\ \dfrac{1}{\tau}||v_{k+1}-v^{n+1}||^2\le\tau\left(g(u_k,v_k)v_{k+1}-g(u^{n+1},v^{n+1})v^{n+1},u_{k+1}-u^{n+1}\right), \end{cases} \tag{2.2.27}$$

where $||\cdot||$ denotes the L_2-norm. We have dropped the superscript $(n+1)$ from the Picard iterates in (2.2.27) for clarity. Concentrating on the equation in the u-variable, we rewrite the RHS as

$$\begin{aligned} &\Big(\big(f(u_k,v_k)-f(u^{n+1},v^{n+1})\big)(u_{k+1}-u^{n+1}),u_{k+1}-u^{n+1}\Big) \\ &\quad+2\Big(\big(f(u_k,v_k)-f(u^{n+1},v^{n+1})\big)u^{n+1},u_{k+1}-u^{n+1}\Big) \\ &\quad+\left(f(u^{n+1},v^{n+1})u_{k+1}-f(u_k,v_k)u^{n+1},u_{k+1}-u^{n+1}\right), \end{aligned} \tag{2.2.28}$$

and obtain

$$\begin{aligned} &\frac{1}{\tau}||u_{k+1}-u^{n+1}||^2 \\ &\quad\le||f(u_k,v_k)-f(u^{n+1},v^{n+1})||\Big(||(u_{k+1}-u^{n+1})^2||+2||u^{n+1}||_\infty||u_{k+1}-u^{n+1}||\Big) \\ &\qquad+\beta_f||u_{k+1}-u^{n+1}||^2 \\ &\quad\le L_f\big(C_1+2||u^{n+1}||_\infty\big)||u_{k+1}-u^{n+1}||\,||\boldsymbol{\xi}_k-\boldsymbol{\xi}^{n+1}||\;+\beta_f||u_{k+1}-u^{n+1}||^2, \end{aligned} \tag{2.2.29}$$

for large enough C_1, and where $\boldsymbol{\xi}_k=(u_k,v_k)$, and similarly for $\boldsymbol{\xi}^{n+1}$. Upon rearrangement, this becomes

$$||u_{k+1}-u^{n+1}||\le L_f\frac{\tau}{1-\tau\beta_f}\big(C_1+2||u^{n+1}||_\infty\big)||\boldsymbol{\xi}_k-\boldsymbol{\xi}^{n+1}||. \tag{2.2.30}$$

Similarly,

$$||v_{k+1}-v^{n+1}||\le L_g\frac{\tau}{1-\tau\beta_g}\big(C_2+2||v^{n+1}||_\infty\big)||\boldsymbol{\xi}_k-\boldsymbol{\xi}^{n+1}||. \tag{2.2.31}$$

Since $||\boldsymbol{\xi}_{k+1}-\boldsymbol{\xi}^{n+1}||\le||u_{k+1}-u^{n+1}||+||v_{k+1}-v^{n+1}||$ we obtain the estimate

$$||\boldsymbol{\xi}_{k+1}-\boldsymbol{\xi}^{n+1}||\le L\frac{\tau}{1-\tau\beta}\big(C+2||\boldsymbol{\xi}^{n+1}||_\infty\big)||\boldsymbol{\xi}_k-\boldsymbol{\xi}^{n+1}||, \tag{2.2.32}$$

where $L = \max\{L_f, L_g\}$, $\beta = \max\{\beta_f, \beta_g\}$, $C = \max\{C_1, C_2\}$. Let $\boldsymbol{\Xi}$ be the exact solution (U, V) to the original problem (2.2.24). Then

$$||\boldsymbol{\xi}^{n+1}||_\infty \leq ||\boldsymbol{\Xi}(t^{n+1}) - \boldsymbol{\xi}^{n+1}||_\infty + ||\boldsymbol{\Xi}(t^{n+1})||_\infty \leq C'(\boldsymbol{\Xi})(h^2 + \tau) + ||\boldsymbol{\Xi}(t^{n+1})||_\infty, \quad (2.2.33)$$

for some constant $C'(\boldsymbol{\Xi})$. Thus, redefining constants as appropriate, we finally obtain

$$\begin{aligned} ||\boldsymbol{\xi}_{k+1} - \boldsymbol{\xi}^{n+1}|| &\leq CL\frac{\tau}{1-\tau\beta}\big(1 + C'(\tau + h^2)\big)||\boldsymbol{\xi}_k - \boldsymbol{\xi}^{n+1}|| \\ &:= \lambda\,||\boldsymbol{\xi}_k - \boldsymbol{\xi}^{n+1}||. \end{aligned} \quad (2.2.34)$$

The coefficient λ will be less than 1 for small enough τ and h, thus showing that the method converges. In particular, $\lambda \to 0$ as $\tau \to 0$ and $h \to 0$ simultaneously.

We can expect linear convergence with the Picard iteration and quadratic convergence with the Newton method (Quarteroni et al. 2006). The Newton method may fail to converge if the initial guess is far from the solution (Quarteroni et al. 2006). However, in our case if we have a small enough timestep, we expect that the solution (u^n, v^n) will provide an adequate initial guess to find the solution (u^{n+1}, v^{n+1}) since the change in the solution through successive timesteps will be small. Evidently, the matrix to be inverted in the Picard system (2.2.13) is symmetric whereas for the Newton system (2.2.21) it is not. This then precludes the use of efficient iterative techniques such as the conjugate gradient method to find a solution to the Newton system. Instead, for the Newton method, we use GMRES in order to solve the non-symmetric system of linear algebraic equations (Saad 1992).

2.3 Computation Implementation

Before we are able to implement an algorithm to solve the Schnakenberg system, we must further discretise the problem. Namely, we must discretise the integrals in the matrices and vectors we set up. To this end, we first rewrite the integrals element-wise, e.g. for the stiffness matrix we rewrite it as

$$A_{ij} = \int_{\Omega_h} \nabla\varphi_i(\boldsymbol{x}) \cdot \nabla\varphi_j(\boldsymbol{x}) = \sum_{\Delta_k} \int_{\Delta_k} \nabla\varphi_i(\boldsymbol{x}) \cdot \nabla\varphi_j(\boldsymbol{x}), \quad (2.3.1)$$

where Δ_k is an element of the mesh. Suppose we have a numerical quadrature rule

$$\int \xi(\boldsymbol{x}) \approx \sum_q w_q(\bar{\boldsymbol{x}_q})\xi(\bar{\boldsymbol{x}_q}), \quad (2.3.2)$$

where ξ is some function to be integrated, the x_q are the quadrature points and w_q are the weights. We approximate the element-wise integral as

$$A_{ij} \approx \sum_{\Delta_k} \sum_{q} w_q(\bar{\boldsymbol{x}}_{\boldsymbol{q}}) \nabla \varphi_i(\bar{\boldsymbol{x}}_{\boldsymbol{q}}) \cdot \nabla \varphi_j(\bar{\boldsymbol{x}}_{\boldsymbol{q}}). \tag{2.3.3}$$

The other integrals can be discretised in a similar fashion.

We use the C++ finite element library *deal.II* (Bangerth et al. 2013) to implement the discretisation described above to be solved on a computer. The library is open-source and provides tools to solve problems that specifically use the finite element method to discretise.

We collect the components of the algorithm into a template class. The class and its public members are as follows:

```
template <int dim, int spacedim>
class SchnakProblem
{
 public:
  SchnakProblem (Parameters &parameters);
  void run ();
 private:
  ...
  void assemble_system();
  ...
};
```

The class takes two template parameters: spacedim is the dimension of the embedding space of the domain and dim is the dimensionality of the domain itself. In this way, the code can be used for a variety of different geometries including solving on surfaces in 3-D and curves in 2-D. This is a feature of the *deal.II* library as many of its classes are also template classes which take the same parameters. The constructor takes in as an argument parameters which the problem needs and the run() function performs the whole algorithm. The details of the implementation are hidden in the private members of the class.

The run() function has the following structure:

```
template <int dim,int spacedim>
void SchnakProblem<dim,spacedim>::run()
{
  // set up function space
```

```
  // create mesh
  // set up degrees of freedom
  while time < maximum time
    {
     // assemble system
     // solve system
     // output solution
     // increment time
    }
}
```

First we choose the finite element function space we wish to work in, which in this case will be piecewise linear polynomials, after which the mesh is created. We choose to use the inbuilt routines in *deal.II* to create meshes of relatively simple geometries, although other meshes can be input by the user. With the finite element space description and the mesh, the global degrees of freedom can be created. Here, each degree of freedom (DoF) is mapped onto the corresponding part of the mesh. In the case of piecewise linear polynomials, each DoF is mapped onto the corresponding vertex of the mesh. This allows the solutions to be stored in a vector-like structure and the matrices to be assembled in a matrix-like structure. The way we constructed the matrices means that they will be sparse matrices. The algorithm will set up the sparsity pattern of the matrices at this point of the algorithm as well. A simple example of what is constructed is shown in Figure 2.1.

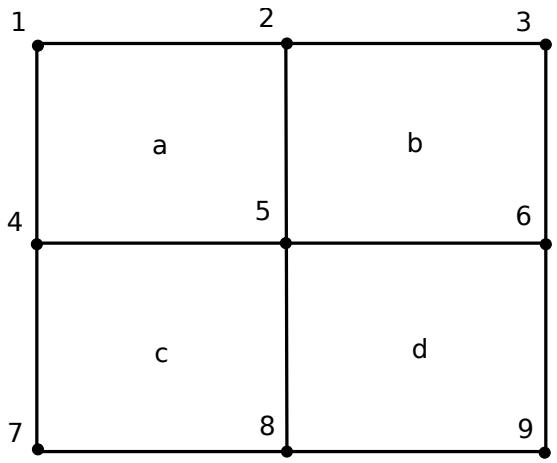


Figure 2.1: A simple example of a mesh of a rectangle using quadrilateral elements. The elements have been lettered *a*-*d* and the vertices have been numbered 1-9.

After this set-up, we are ready to solve the actual problem itself. This is encompassed

in the **while** loop which runs as long as some variable time, which holds the current time of the solution, is less than the a specified maximum time. The system is assembled, i.e. the LHS matrices and the RHS vectors are set up. This is done using the discretisation as in (2.3.3):

```
template <int dim, int spacedim>
void SchnakProblem<dim, spacedim>::assemble_system()
{
 for each element
  {
   for each local DoF
   {
    for each quadrature point
     {
       // get local contributions
     }
   }
   // add local contributions to global containers
  }
}
```

This process can be made to run in parallel since the contributions from each element do not affect each other. Thus, suppose we have n threads. We can divide the elements into n equal shares and have each thread assemble at the same time. This allows for code which runs more quickly.

After assembly the system is solved and is output as necessary.

2.4 Experimental order of convergence (EOC)

There is no known analytical solution to the Schnakenberg system (2.1.1). Therefore, we cannot check directly how well our numerical experiments compare with the exact solution. In the following, we use a well known method of constructing a solution that will satisfy a modified version of (2.1.1) from which we can indirectly gauge the performance of subsequent discretisation. Though this is no substitute for error estimation, it is, nevertheless, a useful guide. The simulations in this section and the next were carried out using the software *deal.II* (Bangerth et al. 2013).

In the following, we take the domain Ω to be the unit square. Define

$$\Xi(x,y,t) = \left(\frac{x^3}{3} - \frac{x^2}{2}\right)\left(\frac{y^3}{3} - \frac{y^2}{2}\right)\left(1 + e^{-t}\right). \tag{2.4.1}$$

Then $u = v = \Xi$ is the exact solution to the modified system

$$\begin{cases} \dfrac{\partial u}{\partial t} - \nabla^2 u - \gamma(-u + u^2 v) = \Xi_t - \nabla^2 \Xi - \gamma(-\Xi + \Xi^3), \\ \dfrac{\partial v}{\partial t} - d\nabla^2 v + \gamma u^2 v = \Xi_t - d\nabla^2 \Xi + \gamma\, \Xi^3, \qquad \text{in } \Omega\,, \text{for } t > 0\,, \end{cases} \tag{2.4.2}$$

with homogeneous Neumann boundary conditions. Initial conditions are then defined by

$$u_0 = v_0 = 2\left(\frac{x^3}{3} - \frac{x^2}{2}\right)\left(\frac{y^3}{3} - \frac{y^2}{2}\right). \tag{2.4.3}$$

Here we shall also take the parameter values $a = 0.1$, $b = 0.9$, $d = 10$ and $\gamma = 29$ (Madzvamuse 2000). It is easily seen that u and v both tend to the inhomogeneous steady $u_0/2$ as $t \to \infty$.

If we now solve (2.4.2) using the proposed time-step methods we can calculate the error from the exact solution Ξ at each timestep. This is done using five different timesteps $\tau_i = 2^{-i}$, $i = 1, 2, 3, 4, 5$ in the time interval $t \in [0, 10]$. In the following, the convergence criteria for the Picard iteration and the Newton method was chosen to be $||u_{k+1}^{n+1} - u_k^{n+1}|| < 10^{-5}$, and similarly for the variable v, where $||\cdot||$ denotes the L_2-norm. A measure of the error from the exact solution at the $(n+1)$-th timestep is given by

$$e_u(t^{n+1}) = ||u^{n+1} - \Xi(t^{n+1})||, \tag{2.4.4}$$

and a measure of the error of the whole simulation is given by

$$E_u = e_u(10), \tag{2.4.5}$$

which is the error at the end time. Similar quantities can be defined for the v-variable. We have the error estimate (Thomée 2006)

$$E_u \leq C[u]\left(h^2 + \tau^\alpha\right), \tag{2.4.6}$$

where $C[u]$ is a constant and h is the greatest length of the squares in the mesh T_h. For a particular time-stepping scheme, the maximal value of α such that the above holds is then its order of convergence. Thus, for BE we expect $\alpha = 1$ and for the CN and the FSTS we expect $\alpha = 2$.

Let $E_{u,i}$ $(i = 1, 2, \cdots, 5)$ denote the errors found as above using the same time-stepping scheme with time step τ_i. The estimate (2.4.6) is a sharp estimate so that if $h^2 \sim \tau^\alpha$, we have approximately

$$\alpha_i \approx \frac{\log(E_{u,i}) - \log(E_{u,i-1})}{\log(\tau_i) - \log(\tau_{i-1})}, \quad i > 1, \tag{2.4.7}$$

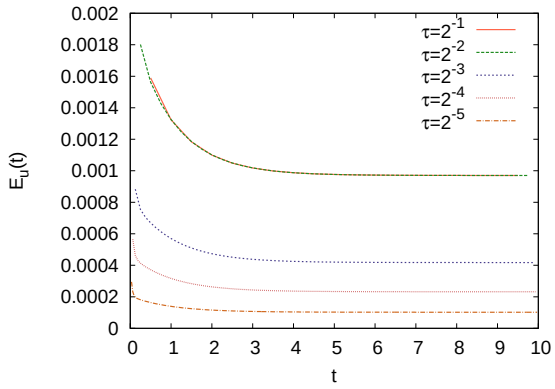
	BE	CN	FSTS
α_1	-1.965×10^{-5}	2.049	2.064
α_2	1.217	2.020	2.020
α_3	0.847	2.006	2.006
α_4	1.184	2.002	2.002
α_5	0.835	-	-
α_6	0.922	-	-
α_7	1.083	-	-
α_8	1.048	-	-
α_9	0.953	-	-

Table 2.1: Experimental order of convergence (EOC) for the u-variable. Further EOC values were computed for BE to investigate the convergence to the expected value of 1.

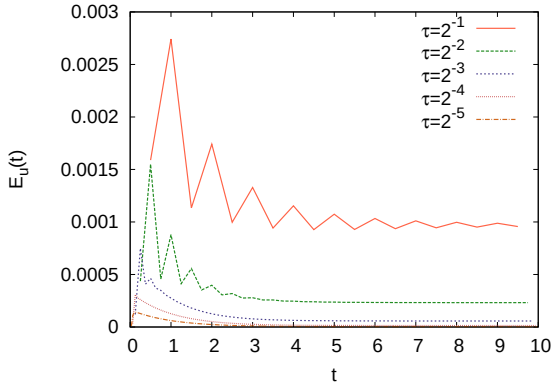
with the α_i converging to the expected value as the errors decrease. For the second order methods, the coupling $h_i = \tau_i$ was obtained by constructing an $n \times n$ square grid with $n = 2^i$. For the backward-Euler method $h_i = \sqrt{\tau_i}$ was taken by constructing a square grid as before but taking the nearest n such that the value of $1/n$ was closest to $\sqrt{\tau_i}$.

The errors for $\tau_5 = 1/32$ are plotted in Fig. 2.2. As expected, the second order methods are more accurate than BE. The convergence results are shown in Table 2.1. The values of α obtained are in keeping with the theory. The apparent non-convergence of the α_i to 1 for BE, due perhaps to the errors not being small enough, prompted further values of α_i to be calculated for BE, keeping the timestep $\tau_i = 2^{-i}$. These are shown in Table 2.1, where the convergence is made more apparent.

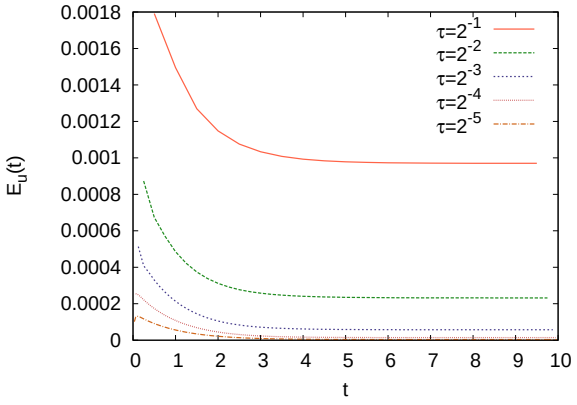
Upon inspection of Fig. 2.2, one can observe oscillations in the error for the CN method which are damped with time. This behaviour is well known (Atkinson 1989, Hughes 2003), and ultimately mars the results. Put briefly, under the CN, high frequency modes of the errors in the initial data do not damp out effectively (Østerby 2003). Some strategies to reduce these oscillations are studied in Britz et al. (2003) and Østerby (2003). Here, we use a strategy in which BE is performed for a few initial timesteps, after which the CN is implemented. Essentially, the BE method damps out the oscillations which are otherwise retained in the CN method (Britz et al. 2003). We try three variants of this method: in the first, we perform BE for the first timestep (CNB); in the second we perform two BE steps (CNB2); and in the third we perform five (CNB5). The results of these implementations are shown in Fig. 2.3 for two timesteps. These methods seem to have varying results. For $\tau = 1/2$, CNB seems to increase the oscillations, whilst CNB5 reduces them the most,



(a)

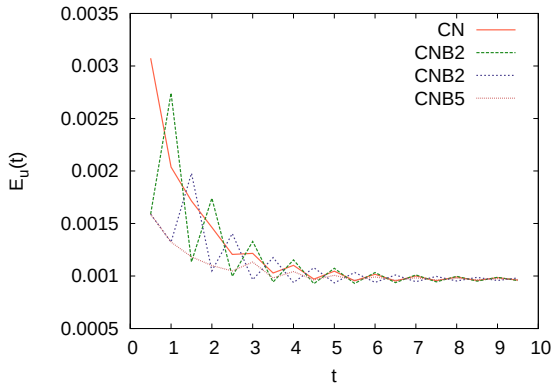


(b)

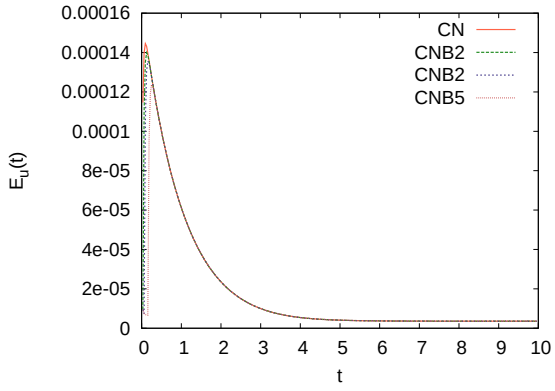


(c)

Figure 2.2: Errors for the modified equation for the u-variable using the adaptive Newton method: (a) BE (b) CN (c) FTS. The second order methods CN and FSTS are more accurate than BE. The error ofconvergence results are shown in Table 2.1.



(a)



(b)

Figure 2.3: Errors in the simulation of the modified equation (2.4.2) using the modified CN methods CNB, CNB2 and CNB5 with (a) $\tau = 1/2$ and (b) $\tau = 1/32$ (see Section 2.4 for details). The oscillations observed in CN are damped by the modified methods for both these timesteps. An adaptive Newton method was employed for the fully implicit time-stepping scheme.

however, for the smaller timestep $\tau = 1/32$, all methods seems to behave similarly. Overall however, the magnitude of the errors are reduced by taking more BE timesteps initially.

2.5 Numerical Results

We have three time-stepping schemes and two ways to solve the non-linearities implicitly. We choose Ω to be the unit square and use a uniform square mesh with $10,000$ squares such that the greatest length is $h = 1/100$. To ensure errors of a similar magnitude, a timestep of $\tau = 10^{-2}$ was used for the CN method and the FSTS and $\tau = 10^{-4}$ was used for the BE method. For the initial conditions, a random perturbation from equilibrium of order $\sim 10^{-2}$ was set on every vertex of the mesh. As before, the convergence criterion for the Picard iteration and the Newton method was chosen to be $||u_{k+1}^{n+1} - u_k^{n+1}|| < 10^{-5}$, and similarly for the variable v.

We expect the system to reach a spatially inhomogeneous steady state. Thus, the simulation was left to run until a maximum time $t = 30$, or until the following criteria were satisfied:

$$\frac{||u^{n+1} - u^n||}{\tau} \leq 10^{-4} \quad \text{and} \quad \frac{||v^{n+1} - v^n||}{\tau} \leq 10^{-4}. \tag{2.5.1}$$

This quantity is related to the rate of change of the variables, and so we simply stopped the calculations when the solutions have stopped varying significantly with time. We find it convenient to divide by the timestep, τ, to allow for better comparison between solutions using different timesteps.

After performing the calculations, we found that the solutions obtained using the Picard iteration and the Newton method were virtually indistinguishable. Table 2.2 shows the number of iterations needed to reach the steady state as defined by (2.5.1). Of note is that the CN method does not reach this steady state. For the values quoted for FSTS, the two linear steps which frame the non-linear step in each timestep are counted. In Fig. 2.4 the number of non-linear iterations performed at each timestep is plotted. The number of Picard iterations varied to a greater degree than those resulting from the Newton method. The number of iterations for the Picard iteration varied between 1 and 7, whereas for the Newton method it varied between 1 and 2.

The convergence history of each time-stepping method using the Newton method is shown in Fig. 2.5. The initial growth of the selected mode is evident, as is the decay of the other modes, afterwhich there is the evolution to the steady state. This behaviour is

(a)

	BE	CN	CNB5	FSTS
end time	18.45	-	18.26	17.91
total iterations	201024	-	4353	3822
elapsed CPU time ($\times 10^4$s)	185.10	25.92	10.45	11.42

(b)

	BE	CN	CNB5	FSTS
end time	18.46	-	18.48	18.14
total iterations	201052	-	3529	2994
elapsed CPU time ($\times 10^4$s)	963.6	105.33	96.8	8.39

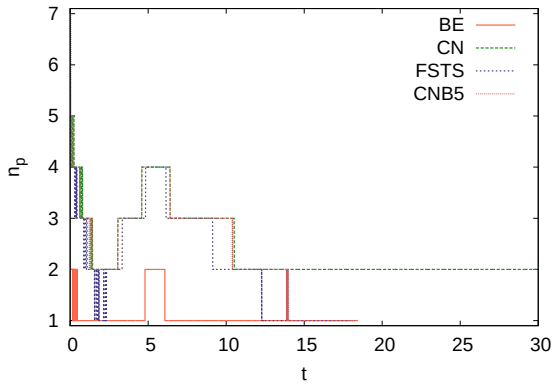
Table 2.2: Details of the simulation of the Schnakenberg system using (a) Picard iteration and (b) the Newton method. Shown are the total number of non-linear iterations required to reach the end time and the elapsed CPU time. Convergence was not obtained with the CN method, therefore the data is not included.

reflected in the number of non-linear iterations per timestep in Fig. 2.4: as the solution is changing to a greater degree between successive timesteps during the exponential growth, more non-linear iterations are needed for convergence. There is good agreement between the solutions using the BE method and FSTS, however there is some discrepancy with the results obtained using the CN and, in fact, the solution using the CN fails to converge in the chosen time limit. From Fig. 2.5 it would seem that there are modes less given to decay in the CN than in the other methods. As seen in Section 2.4, the CN method is prone to oscillations. To address these issues, we shall use the CNB5 method as above where we perform BE for the first five timesteps. The results are shown in Fig. 2.6. As expected, there is much better agreement in this case.

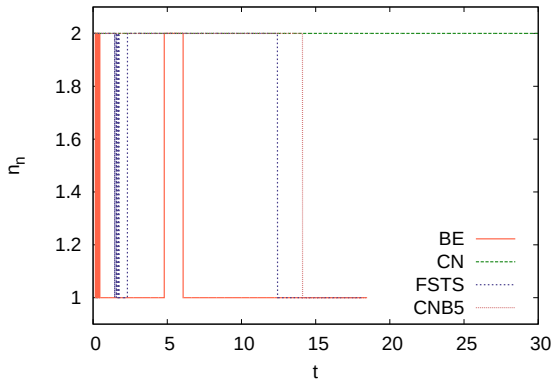
The theoretical exponential growth rate of the excited mode calculated from the linear stability analysis is about $\lambda = 1.6246$. So, if $u \sim e^{\lambda t}$ we should have

$$\frac{||u^{n+1} - u^n||}{||u^n - u^{n-1}||} \sim \frac{e^{\lambda\tau} - 1}{1 - e^{-\lambda\tau}}, \quad n > 2. \tag{2.5.2}$$

The right-hand side (RHS) of (2.5.2) is shown in the straight lines in Fig. 2.7, along with the left-hand side (LHS) for the variable u in each test. The BE and the FSTS methods show excellent agreement with the theory during the period of exponential growth, whereas the CN method shows visible deviation. The results from the modified CNB5 method, presented in Fig. 2.7(b), show better agreement.

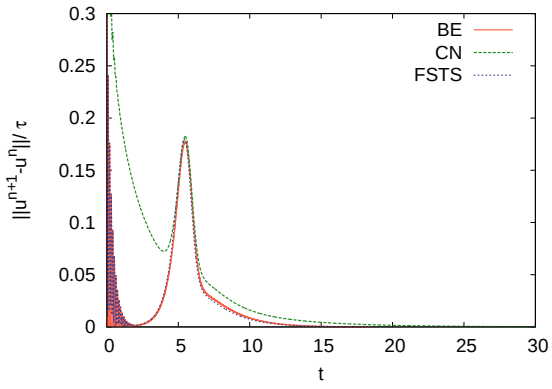


(a)

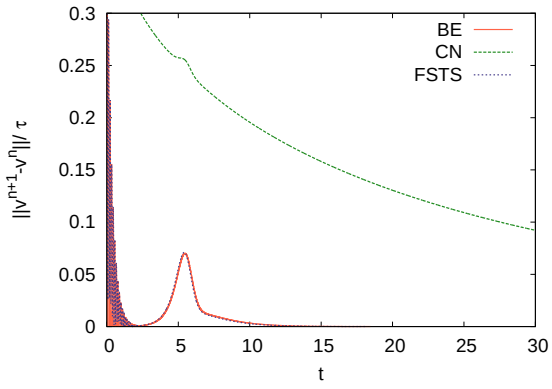


(b)

Figure 2.4: Number of non-linear iterations for (a) Picard iteration, n_p, and (b) the Newton method, n_n, at each timestep.



(a)



(b)

Figure 2.5: Convergence history of the simulation of the Schnakenberg system for (a) the u-variable and (b) the v-variable. Seen is the initial decay of the modes, mode excitation (growth) and decay into the inhomogeneous steady state. An adaptive Newton method was employed for the fully implicit time-stepping scheme.

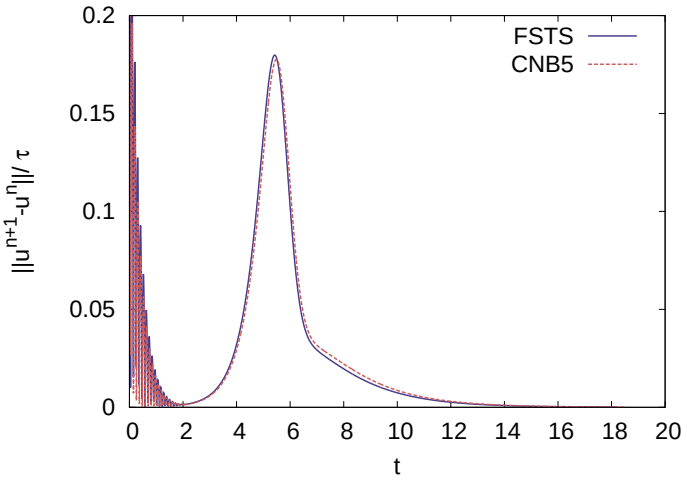
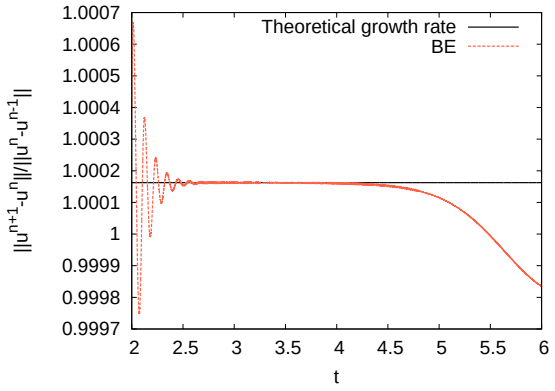
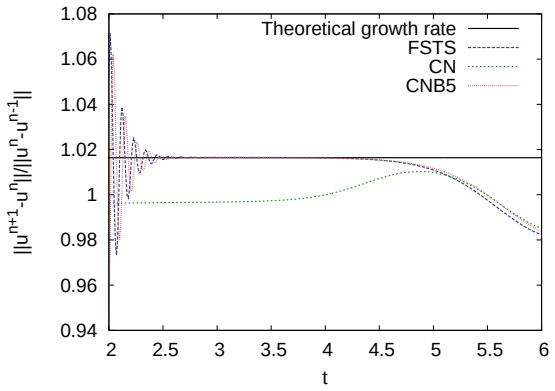


Figure 2.6: Comparison between the convergence history of the FSTS and the modified CNB5 method for the u-variable. Much better agreement is seen here compared to the unmodified CN method. Though not shown, similar improvement is seen in the v-variabe as well. An adaptive Newton method was employed for the fully implicit time-stepping scheme.



(a)



(b)

Figure 2.7: A comparison of growth rates - the horizontal lines correspond to the theoretical growth rate as in the RHS of (2.5.2). (a) BE; (b) CN, CNB5 and FSTS. Excellent agreement during the period of mode excitation is seen between the theoretical growth rate and that obtained in the numerical simulations using BE, CNB5 and FSTS. Using CN, this agreement is lost. An adaptive Newton method was employed for the fully implicit time-stepping scheme.

2.6 Comparison with IMEX schemes

For the sake of comparison, the simulation was carried out without using the iterative techniques to solve the non-linearities but otherwise leaving all other parameters, outlined in Section 2.5, unaltered. Instead, only one Picard iteration was performed at each timestep which is equivalent to linearising the non-linear terms to obtain an IMEX scheme. This was done with BE (BIMEX), CN (CIMEX), CNB5 (C5IMEX) and FSTS (FIMEX). The results are shown in Table 2.3 and Fig. 2.8. The results most affected are the CN schemes, both modified and unmodified. Under the IMEX schemes, neither converges and both suffer from colossal oscillations spanning magnitudes of order 10^2. With FIMEX, the change is not as dramatic, however in Fig. 2.8 the convergence to the steady state towards the end is visibly different. The least affected is BN, with the BIMEX and BN lines in Fig. 2.8 indiscernible from each other. This is to be expected since the time-step τ is smaller for BN and BIMEX than for the second order methods and so the linearisation of the nonlinear terms is more accurate. The solution for CIMEX, C5IMEX and FIMEX are compromised due to the fact that being explicit time-stepping schemes, their region of stability is reduced (Madzvamuse 2006). The value $\tau = 0.01$ does not fall within this region. However, as we saw in Section 2.5, this value of τ gave good results when we treated the non-linearities implicitly. This highlights the fact that in using a fully implicit scheme we can allow ourselves to use a larger timestep thereby allowing for greater numerical stability.

To further investigate this observation, the simulations were carried out again using only one Newton iteration instead of one Picard one. The results agreed well without controlling the number of Newton iterations as in Section 2.5 and the convergence histories

	BIMEX	CIMEX	CBIMEX	FIMEX
end time	18.5024	-	-	19.15
elapsed CPU time ($\times 10^4$s)	153.7	-	-	7.65

Table 2.3: Details of the simulation of the Schnakenberg system using the IMEX schemes with one Picard iteration. Shown are the end time of the simulations and the elapsed CPU time. Convergence was not reached with either the CIMEX or C5IMEX methods, therefore the data is not included.

	BE	CN	CNB5	FSTS
end time	18.4571	-	18.49	18.14
elapsed CPU time ($\times 10^4$s)	793.9	-	87.3	9.97

Table 2.4: Details of the simulation of the Schnakenberg system using the one Newton iteration. Convergence was not obtained with the CN method, therefore the data in not included.

are virtually the same. The information about the end time of simulation and the CPU time taken is shown in Table 2.4. Comparing Tables 2.2 and 2.4 one can see good agreement in the end time of simulation. Hence the time taken by the fully adaptive Newton and the restricted single Newton methods is similar; there does not seem to be much, or any, speed up gained by using the fully adaptive versus the single Newton iteration. This is not at all surprising since the full Newton varied between one and two iterations per timestep, so only taking one iteration does not actually significantly reduce the total elapsed CPU time required.

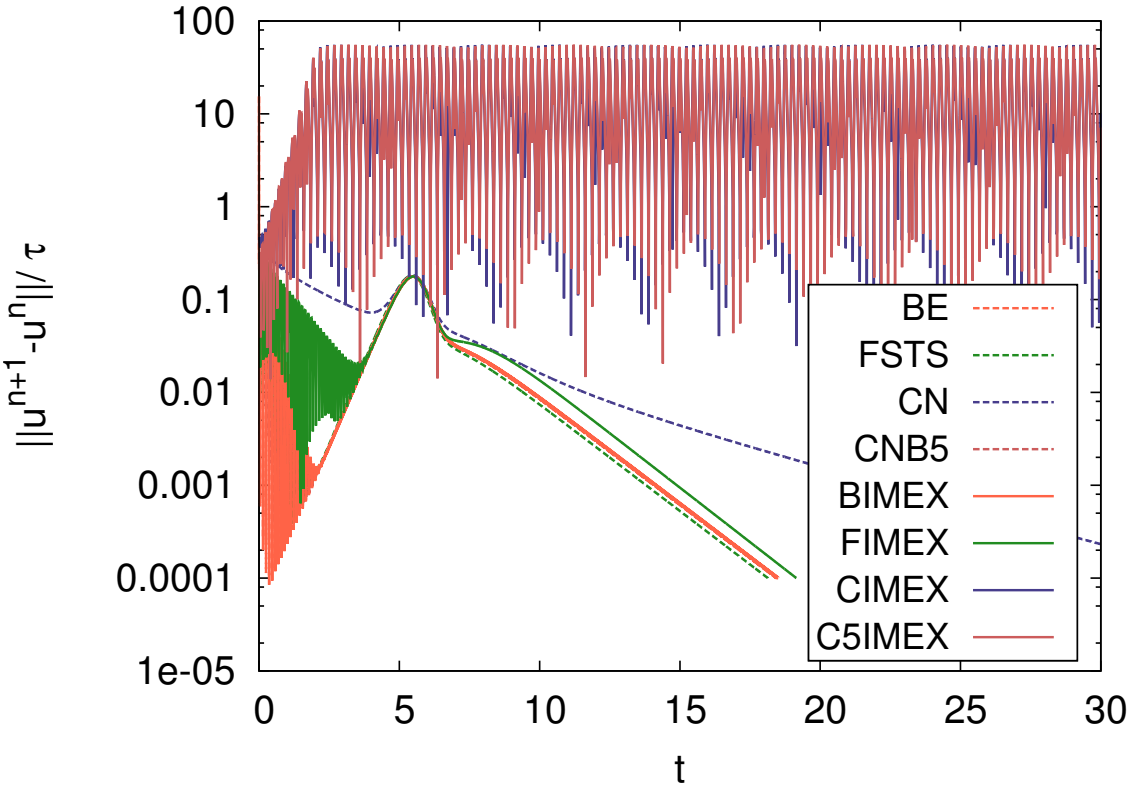


Figure 2.8: Comparison of the convergence history of the simulation of the Schnakenberg system for the IMEX schemes. The solid lines show the IMEX schemes and the dotted lines their fully implicit counterparts.

2.7 Applications to other geometries

The methods outlined are readily applicable to complex geometries in multi-dimensions as well as on surfaces. Here we present some solutions of the Schnakenberg system in the bulk of the unit sphere, the unit cube and on the unit square for different parameter values. For fast and accurate simulations, we use only the FSTS to discretise in time, coupled with the Newton's method (with a single iteration at each timestep) to solve the nonlinearities implicitly. The simulations in this section were carried out using the software *deal.II* (Bangerth et al. 2013).

We choose parameters to isolate modes on the square and use the same parameters on the 3D geometries. As detailed before, for the unit square the eigenmodes of (2.1.7) have the form $\cos(n\pi x)\cos(m\pi y)$ for $n, m \in \mathbb{Z}$ with eigenvalues $k^2 = n^2 + m^2$. We choose the parameter values $a = 0.1$, $b = 0.9$ and firstly $d = 9.1676$, $\gamma = 176.72$, then secondly $d = 8.6076$, $\gamma = 535.09$. The first set isolates modes corresponding to $(n, m) = (2, 1)$, whilst the second isolates those corresponding to $(3, 3)$ (Madzvamuse 2000). The results of such calculations are shown in Fig. 2.9. A wide variety of patterns are obtained in these geometries.

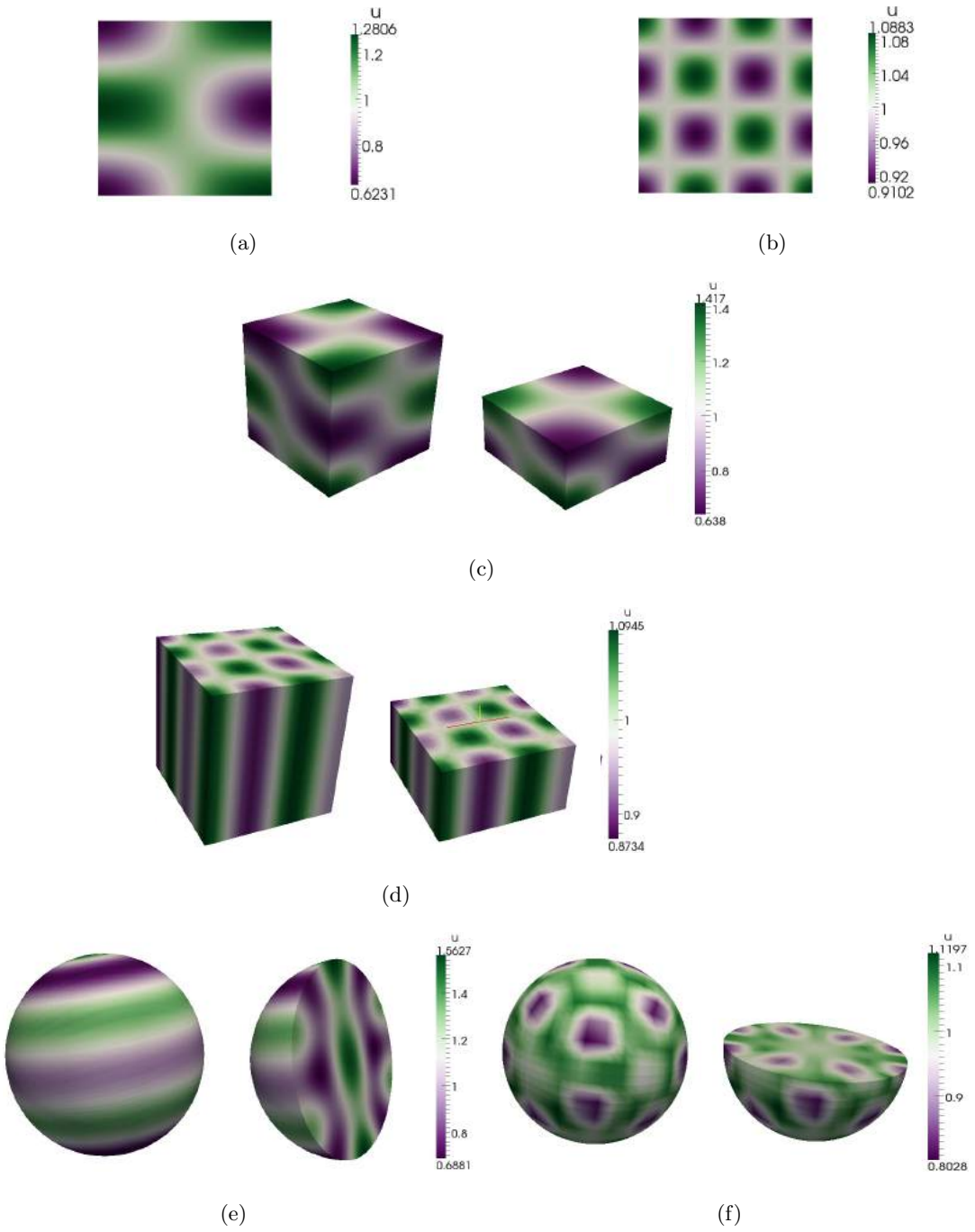


Figure 2.9: Solution for the variable u of the Schnakenberg system on the square (top row), the bulk of the cube (middle row) and the bulk of the sphere (bottom row). Parameter values were $a = 0.1$, $b = 0.9$ and (a), (c), (e) $d = 9.1676$, $\gamma = 176.72$ and (b), (d), (f) $d = 8.6076$, $\gamma = 535.09$. Part of the domain has been cut away and shown on the right in the 3D geometries to reveal some internal structure. A single Newton iteration at each timestep was employed for the fully implicit time-stepping scheme.

2.8 Conclusions

The overall elapsed CPU times of the numerical experiments shown in Table 2.2 demonstrate that the fractional θ-method (FSTS) is about 130 times faster than the backward Euler and 15 times faster than the Crank-Nicholson and its modifications when both the Picard iteration and the Newton method are employed as fully implicit solvers. Furthermore, numerical tests show that a single Newton iteration at each timestep outperforms the Picard iteration. However, there is a drawback; from Table 2.2 it is clear that the Newton method is slower than the Picard iteration in treating non-linearities arising in reaction-diffusion systems when the backward Euler and the Crank-Nicholson method and its modifications are employed. The extra time needed is attributed to the fact that the Newton method to uses the GMRES solver to invert the non-symmetric systems, whereas the Picard iteration uses the CG solver for the symmetric systems which is quicker. In the FSTS method the non-linear system (2.2.11) is slightly different than the others and takes less iterations for the GMRES solver to solve.

The BE scheme and FSTS agreed well with each other, however the CN method had to be modified in order to gain such an agreement. The modified CN method performs very well; it is second order, is more straightforward to implement than the FSTS and is unconditionally stable. However, even though the FSTS does require two extra equations to be solved per timestep, the extra equations are linear and as such do not add any significant amount of work to be done to calculate the solution. As a result, we strongly recommend using the FSTS coupled with the Newton method (with a single iteration at each timestep) as the preferred fully implicit time-stepping scheme for reaction-diffusion systems on stationary and sometimes continuously evolving domains, volumes and surfaces.

The comparison between the IMEX schemes and the fully implicit schemes demonstrate how linearisation affects the results. Solving non-linearities implicitly alleviated the need for smaller timesteps. However, this alleviation came at the price of solving equations that were non-linear. The choice between IMEX and fully implicit schemes then lies in the trade-off between the size of the timestep and the number of extra non-linear iterations at each timestep.

It is to be expected that such results will also apply to other reaction-diffusion systems such as the Gierer-Meinhardt model (Gierer & Meinhardt 1972), the Thomas model

(Thomas & Kernevez 1976) and Brusselator model (Prigogine & Lefever 1968*a*) and indeed reaction-diffusion systems modelling atherosclerosis. In the case of different boundary conditions, the inhomogeneous steady state and the Turing space may change but the same method of discretisation may be used, so that the results presented in this chapter are also applicable.

Although we have presented results for stationary domains, it is natural to extend the numerical analysis of implicit solvers to domains and surfaces that change continuously, i.e. growing domains and evolving surfaces. It is when solving non-autonomous systems of PDEs posed on evolving domains and surfaces (as well as in the bulk) that a single Newton iteration will become crucially important both as an accurate solver as well as an aid for large computational savings. The theory of PDEs posed on evolving domains and surfaces (as well as in the bulk) is a fast burgeoning research area with many applications in cell motility, plant biology and developmental biology where such models are routinely used (Elliott & Ranner 2012, Neilson, Veltman & van Haaster others 2011, Venkataraman et al. 2011, Lakkis et al. 2013).

In this chapter, we have not tried to connect the numerical methods presented with the linear stability analysis shown in Section 2.1. We present such a connection in Appendix A.

The results here can be applied to our example system, atherosclerosis. The onset of atherosclerosis is associated with inflammation as we have seen in Section 1.4. Moreover, a link has been established with the level of inflammation and plaque rupture (Carr et al. 1997, van der Wal & Becker 1999, Shah 2003, Boyle 2005, Stoll & Bendszus 2006). A natural way to model inflammation is through reaction-diffusion equations where the variables can include white blood cells, lipids and foam cells (see Section 1.7). The reaction-diffusion models found in the literature can be solved more efficiently using the methods detailed here.

In the next chapter we shall use the techniques explored here to study a coupled reaction-diffusion system where one system is defined on the bulk of the domain and the other is defined on its surface.

Chapter 3

Bulk-surface coupled RDEs

In this chapter we formulate new models for coupled systems of bulk-surface reaction-diffusion equations on stationary volumes. The bulk reaction-diffusion equations are coupled to the surface reaction-diffusion equations through linear Robin-type boundary conditions. We then state and prove the necessary conditions for diffusion-driven instability for the coupled system. Due to the nature of the coupling between bulk and surface dynamics, we are able to decouple the stability analysis of the bulk and surface dynamics. Under a suitable choice of model parameter values, the bulk reaction-diffusion system can induce patterning on the surface independent of whether or not the surface reaction-diffusion system produces patterning. On the other hand, the surface reaction-diffusion system can not generate patterns everywhere in the bulk in the absence of patterning from the bulk reaction-diffusion system. For this case, patterns can only be induced in regions close to the surface membrane. Various numerical experiments are presented to support our theoretical findings. Our most revealing numerical result is that Robin-type boundary conditions seem to introduce a boundary layer coupling the bulk and surface dynamics.

Such considerations are germane to the mathematical modelling of atherosclerosis. Previously, we have described how reaction-diffusion systems are pertinent in modelling inflammation. Here, we may consider a more complex system. For instance, we could consider a plaque with a thin-fibrous cap and the lipid core which it covers. The lesion may be defined on the bulk geometry whilst the thin-fibrous cap can be described as its surface. With a vulnerable plaque in the carotid artery being taken as one with cap thickness of less than 65μm (Burke et al. 1997), this assumption of a surface does not seem too unreasonable. The interaction of the lipid core and thin-fibrous cap may

then be studied to elucidate mechanisms of rupture relating to inflammation. This line of investigation has to our knowledge not been pursued yet. Whereas reaction-diffusion models do exist of atherogenesis (Khatib et al. 2009, Volpert & Petrovskii 2009, Cilla et al. 2014), none investigate a system specific to the role of inflammation in the rupture of plaque.

3.1 Introduction

In many fluid dynamics applications and biological processes, coupled bulk-surface partial differential equations naturally arise in $(2D + 3D)$. In most of these applications and processes, morphological instabilities occur through symmetry breaking resulting in the formation of heterogeneous distributions of chemical substances (Levine & Rappel 2005). In developmental biology, the emergence and maintenance of polarised states in the form of heterogeneous distributions of chemical substances such as proteins and lipids is essential. Examples of such processes include (but are not limited to) the formation of buds in yeast cells and cell polarisation in biological cells due to responses to external signals through the outer cell membrane (Rätz & Röger 2012, 2014). In the context of reaction-diffusion processes, such symmetry breaking arises when a uniform steady state, stable in the absence of diffusion, is driven unstable when diffusion is present thereby giving rise to the formation of spatially inhomogeneous solutions in a process now well-known as the Turing diffusion-driven instability (Turing 1952). Classical Turing theory requires that one of the chemical species, typically the *inhibitor*, diffuses much faster than the other, the *activator* resulting in what is known as the *long-range inhibition* and *short-range activation* (Gierer & Meinhardt 1972, Murray 2003).

Recently, there has been a surge in studies on models that coupled bulk dynamics to surface dynamics. For example, Rätz & Röger (2014) study symmetry breaking in a bulk-surface reaction-diffusion model for signalling networks. In this work, a single diffusion partial differential equation (the heat equation) is formulated inside the bulk of a cell, while on the cell-surface, a system of two membrane reaction-diffusion equations is formulated. The bulk and cell-surface membrane are coupled through Robin-type boundary conditions and a flux term for the membrane system (Rätz & Röger 2014). Elliott & Ranner (2012) study a finite element approach to a sample elliptic problem: a single elliptic partial differential equation is posed in the bulk and another is posed on the surface. These

are then coupled through Robin-type boundary conditions. Novak et al. (2007) present an algorithm for solving a diffusion equation on a curved surface coupled to a diffusion model in the volume. Chechkin et al. (2012) study bulk-mediated diffusion on planar surfaces. Again, diffusion models are posed in the bulk and on the surface coupling them through boundary conditions. In the area of tissue engineering and regenerative medicine, electrospun membranes are useful in applications such as filtration systems and sensors for chemical detection. Understanding of the fibres' surface, bulk and architectural properties is crucial to the successful development of integrative technology. Nisbet et al. (2009) presents a detailed review on surface and bulk characterisation of electrospun membranes of porous and fibrous polymer materials. To explain the long-range proton translocation along biological mombranes, Medvdev & Stuchebrukhov (2013) propose a model that takes into account the coupled bulk-diffusion that accompanies the migration of protons on the surface. More recently, Rozada et al. (2014) presented singular perturbation theory for the stability of localised spot patterns for the Brusselator model on the sphere.

In most of the work above, either elliptic or diffusion models in the bulk have been coupled to surface-elliptic or surface-diffusion or surface-reaction-diffusion models posed on the surface through Robin-type boundary conditions (Chechkin et al. 2012, Elliott et al. 2013, Medvdev & Stuchebrukhov 2013, Nisbet et al. 2009, Novak et al. 2007, Rätz & Röger 2012, 2014). Here, our focus is to couple systems of reaction-diffusion equations posed both in the bulk and on the surface, setting a mathematical and computational framework to study more complex interactions such as those observed in cell biology, tissue engineering and regenerative medicine, developmental biology and biopharmaceuticals (Chechkin et al. 2012, Elliott et al. 2013, Medvdev & Stuchebrukhov 2013, Nisbet et al. 2009, Novak et al. 2007, Rätz & Röger 2012, 2014, Venkataraman et al. 2011). We employ the bulk-surface finite element method as introduced by Elliott & Ranner (2012) to numerically solve the coupled system of bulk-surface reaction-diffusion equations. Details of the surface-finite element can be found in Dziuk & Elliott (2007). The bulk and surface reaction-diffusion systems are coupled through Robin-type boundary conditions. The coupled bulk-surface finite element algorithm is implemented in the C++ library *deal.II* (Bangerth et al. 2013).

The key contributions of our work to the theory of pattern formation are:

- We derive Turing diffusion-driven instability conditions for a coupled system of bulk-surface reaction-diffusion equations.

- Using a bulk-surface finite element method, we approximate the solution to the model system within the bulk and on the boundary surface of a sphere of radius one.

- Our results show that if the surface-reaction-diffusion system has the *long-range inhibition, short-range activation* form and the bulk-reaction-diffusion system has equal diffusion coefficients, then the surface-reaction-diffusion system can induce patterns in the bulk close to the surface and no patterns form in the interior, far away from the surface.

- On the other hand, if the bulk-reaction-diffusion system has the *long-range inhibition, short-range activation* form and the surface-reaction-diffusion system has equal diffusion coefficients, then the bulk-reaction-diffusion system can induce pattern formation on the surface.

- Furthermore, we prove that if the bulk and surface reaction-diffusion systems have equal diffusion coefficients, no patterns form.

- These theoretical predictions are supported by numerical simulations.

This chapter is outlined as follows. In Section 3.2 we present the coupled bulk-surface reaction-diffusion system on stationary volumes with appropriate boundary conditions coupling the bulk and surface partial differential equations. The main results of this chapter are presented in Section 3.2.2 where we derive Turing diffusion-driven instability conditions for the coupled system of bulk-surface reaction-diffusion equations. To validate our theoretical findings, we present bulk-surface finite element numerical solutions in Section 3.4. In Section 3.6, we conclude and discuss the implications of our findings.

3.2 Coupled bulk-surface reaction-diffusion systems on stationary volumes

In this section we present a coupled system of bulk-surface reaction-diffusion equations (BSRDEs) posed in a three-dimensional volume as well as on the boundary surface enclosing the volume. We impose Robin-type boundary conditions on the bulk reaction-diffusion system while no boundary conditions are imposed on the surface reaction-diffusion system since the surface is closed.

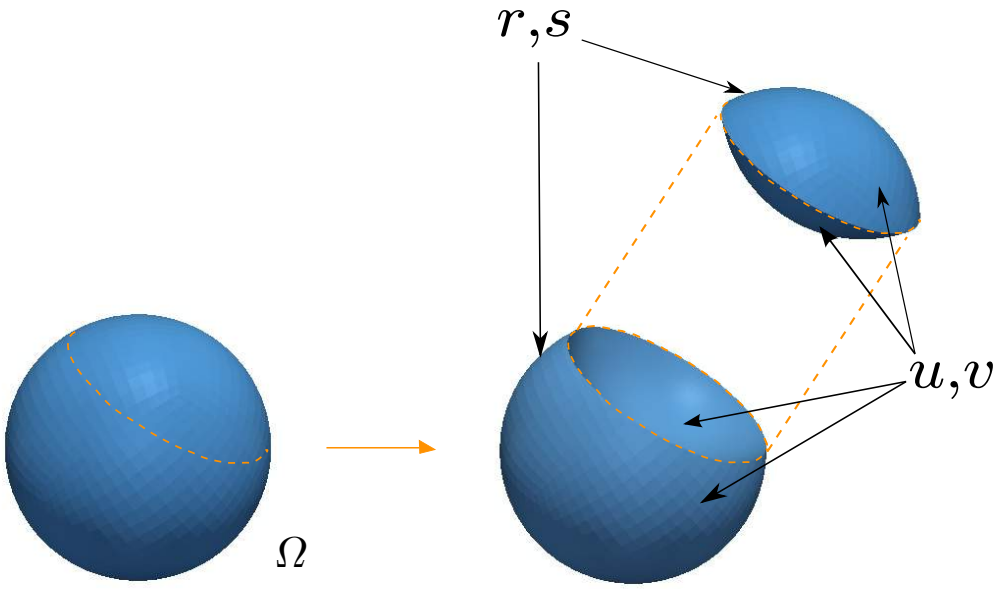


Figure 3.1: The variables of the bulk-surface coupled system. On the left is shown the whole bulk domain Ω. We make a cut along the dotted line and show an exploded view on the right. The variables u and v are defined both in the interior and boundary of Ω, while the variables r and s are only defined on the boundary.

3.2.1 A coupled system of bulk-surface reaction-diffusion equations (BSRDEs)

Let Ω be a stationary volume (whose interior is denoted the bulk) enclosed by a compact hypersurface $\Gamma := \partial\Omega$ which is C^2. Also, let $I = [0, T]$ $(T > 0)$ be some time interval. Moreover, let $\boldsymbol{\nu}$ denote the unit outer normal to Γ, and let U be any open subset of $\mathbb{R}^{N+1}$ containing Γ, then for any function u which is differentiable in U, we define the tangential gradient on Γ by, $\nabla_\Gamma u = \nabla u - (\nabla u \cdot \boldsymbol{\nu})\boldsymbol{\nu}$, where $\cdot$ denotes the regular dot product and ∇ denotes the regular gradient in $\mathbb{R}^{N+1}$. The tangential gradient is the projection of the regular gradient onto the tangent plane, thus $\nabla_\Gamma u \cdot \boldsymbol{\nu} = 0$. The Laplace-Beltrami operator on the surface Γ is then defined to be the tangential divergence of the tangential gradient $\Delta_\Gamma u = \nabla_\Gamma \cdot \nabla_\Gamma u$. For a vector function $\boldsymbol{u} = (u_1, u_2, \ldots, u_{N+1}) \in \mathbb{R}^{N+1}$ the tangential divergence is defined by

$$\nabla_\Gamma \cdot \boldsymbol{u} = \nabla \cdot \boldsymbol{u} - \sum_{i=1}^{N+1} \Big(\nabla u_i \cdot \boldsymbol{\nu}\Big)\nu_i.$$

To proceed, we denote by $u : \Omega \times I \to \mathbb{R}$ and $v : \Omega \times I \to \mathbb{R}$ two chemical concentrations (species) that react and diffuse in Ω and and $r : \Gamma \times I \to \mathbb{R}$ and $s : \Gamma \times I \to \mathbb{R}$ be two chemical species residing only on the surface Γ which react and diffuse on the surface (see Fig. 3.1). In the absence of cross-diffusion and assuming that coupling is only through the

reaction kinetics, we propose to study the following non-dimensionalised coupled system of BSRDEs

$$\begin{cases} \begin{cases} u_t = \nabla^2 u + \gamma_\Omega f(u,v), \\ v_t = d_\Omega \nabla^2 v + \gamma_\Omega g(u,v), \end{cases} & \text{in } \Omega \times (0,T], \\ \\ \begin{cases} r_t = \nabla_\Gamma^2 r + \gamma_\Gamma \Big(f(r,s) - h_1(u,v,r,s) \Big), \\ s_t = d_\Gamma \nabla_\Gamma^2 s + \gamma_\Gamma \Big(g(r,s) - h_2(u,v,r,s) \Big), \end{cases} & \text{on } \Gamma \times (0,T], \end{cases} \tag{3.2.1}$$

with coupling boundary conditions

$$\begin{cases} \frac{\partial u}{\partial \boldsymbol{\nu}} &= \gamma_\Gamma h_1(u,v,r,s), \\ d_\Omega \frac{\partial v}{\partial \boldsymbol{\nu}} &= \gamma_\Gamma h_2(u,v,r,s), \end{cases} \quad \text{on } \Gamma \times (0,T]. \tag{3.2.2}$$

In the above, $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$ represents the Laplacian operator. d_Ω and d_Γ are positive diffusion coefficients in the bulk and on the surface respectively, representing the ratio between u and v, and r and s, respectively. γ_Ω and γ_Γ represent the length scale parameters in the bulk and on the surface respectively. In this formulation, we assume that $f(\cdot,\cdot)$ and $g(\cdot,\cdot)$ are nonlinear reaction kinetics in the bulk and on the surface. $h_1(u,v,r,s)$ and $h_2(u,v,r,s)$ are reactions representing the coupling of the internal dynamics in the bulk Ω to the surface dynamics on the surface Γ. Inspired by Macdonald et al. (2013), we will, as a first attempt, consider a more generalised form of their linear coupling of the following nature

$$h_1(u,v,r,s) = \alpha_1 r - \beta_1 u - \kappa_1 v, \tag{3.2.3}$$

$$h_2(u,v,r,s) = \alpha_2 s - \beta_2 u - \kappa_2 v, \tag{3.2.4}$$

where α_1, α_2, β_1, β_2, κ_1 and κ_2 are constant non-dimensionalised parameters. Initial conditions are given by the positive bounded functions $u_0(\boldsymbol{x})$, $v_0(\boldsymbol{x})$, $r_0(\boldsymbol{x})$ and $s_0(\boldsymbol{x})$.

Activator-depleted reaction kinetics: An illustrative example

From now onwards, we restrict our analysis and simulations to the well-known *activator-depleted* substrate reaction model (Gierer & Meinhardt 1972, Lakkis et al. 2013, Prigogine & Lefever 1968*b*, Schnakenberg 1979, Venkataraman et al. 2012) also known as the Brusselator given by

$$f(u,v) = a - u + u^2 v, \quad \text{and} \quad g(u,v) = b - u^2 v, \tag{3.2.5}$$

where a and b are positive parameters. For notational simplicity, we postulate the model system (3.2.1) in a more compact form given by

$$\begin{cases} \begin{cases} u_t = \nabla^2 u + f_1(u,v,r,s), \\ v_t = d_\Omega \nabla^2 v + f_2(u,v,r,s), \end{cases} & \boldsymbol{x} \text{ on } \Omega,\ t>0, \\ \\ \begin{cases} r_t = \nabla_\Gamma^2 r + f_3(u,v,r,s), \\ s_t = d_\Gamma \nabla_\Gamma^2 s + f_4(u,v,r,s), \end{cases} & \boldsymbol{x} \text{ on } \Gamma,\ t>0, \end{cases} \tag{3.2.6}$$

with coupling boundary conditions (3.2.2)-(3.2.4). In the above, we have defined appropriately

$$f_1(u,v,r,s) = \gamma_\Omega(a - u + u^2 v), \tag{3.2.7}$$

$$f_2(u,v,r,s) = \gamma_\Omega(b - u^2 v), \tag{3.2.8}$$

$$f_3(u,v,r,s) = \gamma_\Gamma\big(a - r + r^2 s - \alpha_1 r + \beta_1 u + \kappa_1 v\big), \tag{3.2.9}$$

$$f_4(u,v,r,s) = \gamma_\Gamma\big(b - r^2 s - \alpha_2 s + \beta_2 u + \kappa_2 v\big). \tag{3.2.10}$$

3.2.2 Linear stability analysis of the coupled system of BSRDEs

Definition 3.2.1 (Uniform steady state)**.** *A point (u^*, v^*, r^*, s^*) is a uniform steady state of the coupled system of BSRDEs (3.2.6) with reaction kinetics (3.2.5) if it solves the nonlinear algebraic system given by $f_i(u^*, v^*, r^*, s^*) = 0$, for all $i = 1, 2, 3, 4$, and satisfies the boundary conditions given by (3.2.2)-(3.2.4).*

Proposition 3.2.1 (Existence and uniqueness of the uniform steady state)**.** *The coupled system of BSRDEs (3.2.6) with boundary conditions (3.2.2) admits a unique steady state given by*

$$(u^*, v^*, r^*, s^*) = \left(a+b, \frac{b}{(a+b)^2}, a+b, \frac{b}{(a+b)^2}\right), \tag{3.2.11}$$

provided the following compatibility conditions on the coefficients of the coupling are satisfied

$$\alpha_1 - \beta_1 = \kappa_1 \frac{b}{(a+b)^3} \tag{3.2.12}$$

$$\beta_2 = (\alpha_2 - \kappa_2)\frac{b}{(a+b)^3}\,. \tag{3.2.13}$$

Furthermore, if we have both $\kappa_1 \neq 0$ and $\alpha_2 \neq \kappa_2$ then the compatibility condition becomes

$$(\beta_1 - \alpha_1)(\kappa_2 - \alpha_2) - \kappa_1 \beta_2 = 0. \tag{3.2.14}$$

Proof. The proof follows immediately from the definition of the uniform steady state satisfying reaction kinetics (3.2.7)-(3.2.10). It must be noted that in deriving this unique uniform steady state, the compatibility conditions above coupling bulk and surface dynamics must be satisfied. □

Remark 3.2.1. Note that there exists an infinite number of solutions to problem (3.2.14).

Linear stability analysis in the absence of diffusion

Next, we study Turing diffusion-driven instability for the coupled system of BSRDEs (3.2.1)-(3.2.4) with reaction kinetics (3.2.5). To proceed, we first consider the linear stability of the spatially uniform steady state. For convenience's sake, let us denote by $\boldsymbol{w} = \Big(u,\ v,\ r,\ s\Big)^T$, the vector of the species u, v, r and s. Furthermore, defining the vector $\boldsymbol{\xi}$ such that $|\xi_i| \ll 1$ for all $i = 1,\ 2,\ 3$ and 4, it follows that writing $\boldsymbol{w} = \boldsymbol{w}^* + \boldsymbol{\xi}$, the linearized system of coupled BSRDEs can be posed as

$$\boldsymbol{w}_t = \boldsymbol{\xi}_t = \boldsymbol{J_F}\boldsymbol{\xi}, \tag{3.2.15}$$

where $\boldsymbol{J_F}$ represents the Jacobian matrix representing the first linear terms of the linearization process. Its entries are defined by

$$\begin{aligned}\boldsymbol{J_F} &= \begin{pmatrix} \frac{\partial f_1}{\partial u} & \frac{\partial f_1}{\partial v} & \frac{\partial f_1}{\partial r} & \frac{\partial f_1}{\partial s} \\ \frac{\partial f_2}{\partial u} & \frac{\partial f_2}{\partial v} & \frac{\partial f_2}{\partial r} & \frac{\partial f_2}{\partial s} \\ \frac{\partial f_3}{\partial u} & \frac{\partial f_3}{\partial v} & \frac{\partial f_3}{\partial r} & \frac{\partial f_3}{\partial s} \\ \frac{\partial f_4}{\partial u} & \frac{\partial f_4}{\partial v} & \frac{\partial f_4}{\partial r} & \frac{\partial f_4}{\partial s} \end{pmatrix} = \begin{pmatrix} f_{1u} & f_{1v} & 0 & 0 \\ f_{2u} & f_{2v} & 0 & 0 \\ f_{3u} & f_{3v} & f_{3r} & f_{3s} \\ f_{4u} & f_{4v} & f_{4r} & f_{4s} \end{pmatrix} \\ &:= \begin{pmatrix} f_u & f_v & 0 & 0 \\ g_u & g_v & 0 & 0 \\ -h_{1u} & -h_{1v} & f_r - h_{1r} & f_s - h_{1s} \\ -h_{2u} & -h_{2v} & g_r - h_{2r} & g_s - h_{2s} \end{pmatrix}. \end{aligned} \tag{3.2.16}$$

where by definition $f_{1u} := \frac{\partial f_1}{\partial u}$ represents a partial derivative of $f_1(u, v)$ with respect to u. We are looking for solutions to the system of linear ordinary differential equations (3.2.15) which are of the form $\boldsymbol{\xi} \propto e^{\lambda t}$. Substituting into (3.2.15), results in the following classical eigenvalue problem

$$\Big|\lambda \boldsymbol{I} - \boldsymbol{J_F}\Big| = 0, \tag{3.2.17}$$

where $\boldsymbol{I}$ is the identity matrix. Making appropriate substitutions and carrying out standard calculations we obtain the following dispersion relation for λ

$$\left|\lambda \boldsymbol{I} - \boldsymbol{J_F}\right| = \begin{vmatrix} \lambda - f_{1u} & f_{1v} & 0 & 0 \\ f_{2u} & \lambda - f_{2v} & 0 & 0 \\ f_{3u} & f_{3v} & \lambda - f_{3r} & f_{3s} \\ f_{4u} & f_{4v} & f_{4r} & \lambda - f_{4s}, \end{vmatrix} = 0, \tag{3.2.18}$$

which can be written

$$\left(\lambda^2 - (f_{1u} + f_{2v})\lambda + f_{1u}f_{2v} - f_{1v}f_{2u}\right)\left(\lambda^2 - (f_{3r} + f_{4s})\lambda + f_{3r}f_{4s} - f_{3s}f_{4r}\right) = 0. \tag{3.2.19}$$

This clearly shows coupling of the bulk and surface dispersion relations in the absence of spatial variations. For convenience's sake, let us denote by

$$\left(\boldsymbol{J_F}\right)_\Omega := \begin{pmatrix} f_{1u} & f_{1v} \\ f_{2u} & f_{2v} \end{pmatrix} \quad \text{and} \quad \left(\boldsymbol{J_F}\right)_\Gamma := \begin{pmatrix} f_{3r} & f_{3s} \\ f_{4r} & f_{4s} \end{pmatrix} \tag{3.2.20}$$

the submatrices of $\boldsymbol{J_F}$ corresponding to the bulk reaction kinetics and the surface reaction kinetics respectively. We can now define

$$\mathrm{Tr}\left(\boldsymbol{J_F}\right) := f_{1u} + f_{2v} + f_{3r} + f_{4s}, \qquad \mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Omega := f_{1u} + f_{2v}, \quad \mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Gamma := f_{3r} + f_{4s},$$
$$\mathrm{Det}\left(\boldsymbol{J_F}\right)_\Omega := f_{1u}f_{2v} - f_{1v}f_{2u}, \quad \text{and} \quad \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Gamma := f_{3r}f_{4s} - f_{3s}f_{4r},$$

and rewrite(3.2.19) as

$$\left(\lambda^2 - \mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Omega \lambda + \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Omega\right)\left(\lambda^2 - \mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Gamma \lambda + \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Gamma\right) = 0 \tag{3.2.21}$$

which expands to give

$$p_4(\lambda) := a_4\lambda^4 + a_3\lambda^3 + a_2\lambda^2 + a_1\lambda + a_0 = 0, \tag{3.2.22}$$

where the coefficients are

$$a_0 = \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Omega \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Gamma, \tag{3.2.23}$$
$$a_1 = -\left[\mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Omega \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Gamma + \mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Gamma \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Omega\right], \tag{3.2.24}$$
$$a_2 = \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Omega + \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Gamma + \mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Omega \mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Gamma, \tag{3.2.25}$$
$$a_3 = -\mathrm{Tr}\left(\boldsymbol{J_F}\right), \tag{3.2.26}$$
$$a_4 = 1. \tag{3.2.27}$$

Theorem 3.2.1 (Necessary and sufficient conditions for $\mathrm{Re}(\lambda) < 0$)**.** *The necessary and sufficient conditions such that the zeros of the polynomial $p_4(\lambda)$ have $Re(\lambda) < 0$ are given by the following conditions*

$$Det\left(\boldsymbol{J_F}\right)_\Omega Det\left(\boldsymbol{J_F}\right)_\Gamma > 0, \tag{3.2.28}$$

$$Tr\left(\boldsymbol{J_F}\right)_\Omega Det\left(\boldsymbol{J_F}\right)_\Gamma + Tr\left(\boldsymbol{J_F}\right)_\Gamma Det\left(\boldsymbol{J_F}\right)_\Omega < 0, \tag{3.2.29}$$

$$Det\left(\boldsymbol{J_F}\right)_\Omega + Det\left(\boldsymbol{J_F}\right)_\Gamma + Tr\left(\boldsymbol{J_F}\right)_\Omega Tr\left(\boldsymbol{J_F}\right)_\Gamma > 0, \tag{3.2.30}$$

$$Tr\left(\boldsymbol{J_F}\right) < 0, \tag{3.2.31}$$

$$\begin{aligned}&\Big[Tr\left(\boldsymbol{J_F}\right)_\Omega Det\left(\boldsymbol{J_F}\right)_\Gamma + Tr\left(\boldsymbol{J_F}\right)_\Gamma Det\left(\boldsymbol{J_F}\right)_\Omega\Big] \\ &\quad - Tr\left(\boldsymbol{J_F}\right)\Big[Det\left(\boldsymbol{J_F}\right)_\Omega + Det\left(\boldsymbol{J_F}\right)_\Gamma + Tr\left(\boldsymbol{J_F}\right)_\Omega Tr\left(\boldsymbol{J_F}\right)_\Gamma\Big] > 0,\end{aligned} \tag{3.2.32}$$

$$\begin{aligned}&\Big[Tr\left(\boldsymbol{J_F}\right)_\Omega Det\left(\boldsymbol{J_F}\right)_\Gamma + Tr\left(\boldsymbol{J_F}\right)_\Gamma Det\left(\boldsymbol{J_F}\right)_\Omega\Big]\Big[Tr\left(\boldsymbol{J_F}\right)\Big(Det\left(\boldsymbol{J_F}\right)_\Omega \\ &\quad + Det\left(\boldsymbol{J_F}\right)_\Gamma + Tr\left(\boldsymbol{J_F}\right)_\Omega Tr\left(\boldsymbol{J_F}\right)_\Gamma\Big)\Big] \\ &\quad - \Big[Tr\left(\boldsymbol{J_F}\right)_\Omega Det\left(\boldsymbol{J_F}\right)_\Gamma + Tr\left(\boldsymbol{J_F}\right)_\Gamma Det\left(\boldsymbol{J_F}\right)_\Omega\Big]^2 \\ &\quad - Tr\left(\boldsymbol{J_F}\right)^2 Det\left(\boldsymbol{J_F}\right)_\Omega Det\left(\boldsymbol{J_F}\right)_\Gamma > 0.\end{aligned} \tag{3.2.33}$$

Proof. The proof enforces that $p_4(\lambda)$ is a Hurwitz polynomial and therefore satisfies the Routh-Hurwitz criterion in order for $\mathrm{Re}(\lambda) < 0$. For the fourth-order polynomial (3.2.22), the Routh-Hurwitz criterion then reads:

$$a_i > 0, \tag{3.2.34}$$

$$a_3 a_2 > a_4 a_1, \tag{3.2.35}$$

$$a_3 a_2 a_1 > a_4 a_1^2 + a_3^2 a_0, \tag{3.2.36}$$

for $i = 0, \ldots, 4$. These result in the conditions given in the theorem. □

Linear stability analysis in the presence of diffusion

Next we introduce spatial variations and study under what conditions the uniform steady state is linearly unstable. We linearize around the uniform steady state by taking small spatially varying perturbations of the form

$$\boldsymbol{w}(\boldsymbol{x}, t) = \boldsymbol{w}^* + \epsilon \boldsymbol{\xi}(\boldsymbol{x}, t), \quad \text{with} \quad \epsilon \ll 1. \tag{3.2.37}$$

Substituting (3.2.37) into the coupled system of BSRDEs (3.2.1)-(3.2.4) with reaction kinetics (3.2.5) we obtain a linearized system of partial differential equations

$$\xi_{1t} = \nabla^2\xi_1 + \gamma_\Omega\Big(f_u\xi_1 + f_v\xi_2\Big), \tag{3.2.38}$$

$$\xi_{2t} = d_\Omega\nabla^2\xi_2 + \gamma_\Omega\Big(g_u\xi_1 + g_v\xi_2\Big), \tag{3.2.39}$$

$$\xi_{3t} = \nabla_\Gamma^2\xi_3 + \gamma_\Gamma\Big(f_r\xi_3 + f_s\xi_4 - h_{1u}\xi_1 - h_{1v}\xi_2 - h_{1r}\xi_3 - h_{1s}\xi_4\Big), \tag{3.2.40}$$

$$\xi_{4t} = d_\Gamma\nabla_\Gamma^2\xi_4 + \gamma_\Gamma\Big(g_r\xi_3 + g_s\xi_4 - h_{2u}\xi_1 - h_{2v}\xi_2 - h_{2r}\xi_3 - h_{2s}\xi_4\Big), \tag{3.2.41}$$

with linearised boundary conditions

$$\frac{\partial\xi_1}{\partial\boldsymbol{\nu}} = \gamma_\Gamma\Big(h_{1u}\xi_1 + h_{1v}\xi_2 + h_{1r}\xi_3 + h_{1s}\xi_4\Big), \tag{3.2.42}$$

$$d_\Gamma\frac{\partial\xi_2}{\partial\boldsymbol{\nu}} = \gamma_\Gamma\Big(h_{2u}\xi_1 + h_{2v}\xi_2 + h_{2r}\xi_3 + h_{2s}\xi_4\Big). \tag{3.2.43}$$

In the above, we have used the original reaction kinetics for the purpose of clarity.

In order to proceed, we restrict our analysis to circular and spherical domains where we can transform the cartesian coordinates into polar and spherical polar coordinates and exploit the method of separation of variables. Without loss of generality, we write the following eigenvalue problem in the bulk

$$\nabla^2\psi_{k_{l,m}}(r) = -k_{l,m}^2\psi_{k_{l,m}}(r), \quad 0 < r < 1, \tag{3.2.44}$$

where each ψ_k satisfies the boundary conditions (3.2.42) and (3.2.43). On the surface the eigenvalue problem is posed as

$$\nabla_\Gamma^2\phi(y) = -l(l+1)\phi(y), \quad y \in \Gamma. \tag{3.2.45}$$

Remark 3.2.2. For the case of circular and spherical domains, if $r = 1$, then $k_{l,m}^2 = l(l+1)$.

Taking $x \in \mathbb{B}$, $y \in \Gamma$, then writing in polar coordinates $x = ry$, $r \in (0,1)$ we can define, for all $l \in \mathbb{N}_0$, $m \in \mathbb{Z}$, $|m| \leq l$, the following power series solutions (Rätz & Röger 2012, 2014)

$$\xi_1(ry,t) = \sum u_{l,m}(t)\psi_{k_{l,m}}(r)\phi_{l,m}(y), \quad \xi_2(ry,t) = \sum v_{l,m}(t)\psi_{k_{l,m}}(r)\phi_{l,m}(y), \tag{3.2.46}$$

$$\xi_3(y,t) = \sum r_{l,m}(t)\phi_{l,m}(y), \quad \text{and} \quad \xi_4(y,t) = \sum s_{l,m}(t)\phi_{l,m}(y). \tag{3.2.47}$$

On the surface, substituting the power series solutions (3.2.47) into (3.2.40) and (3.2.41)

we have

$$\frac{dr_{l,m}}{dt} = -l(l+1)r_{l,m} + \gamma_\Gamma\Big(f_r r_{l,m} + f_s s_{l,m}\Big) - \gamma_\Gamma\Big(h_{1u}u_{l,m}\psi_{k_{l,m}}(1) + h_{1v}v_{l,m}\psi_{k_{l,m}}(1) + h_{1r}r_{l,m} + h_{1s}s_{l,m}\Big), \tag{3.2.48}$$

$$\frac{ds_{l,m}}{dt} = -d_\Gamma l(l+1)s_{l,m} + \gamma_\Gamma\Big(g_r r_{l,m} + g_s s_{l,m}\Big) - \gamma_\Gamma\Big(h_{2u}u_{l,m}\psi_{k_{l,m}}(1) + h_{2v}v_{l,m}\psi_{k_{l,m}}(1) + h_{2r}r_{l,m} + h_{2s}s_{l,m}\Big). \tag{3.2.49}$$

Similarly, substituting the power series solutions (3.2.46) into the bulk equations (3.2.38) and (3.2.39) we obtain the following system of ordinary differential equations

$$\frac{du_{l,m}}{dt} = -k_{l,m}^2 u_{l,m} + \gamma_\Omega\Big(f_u u_{l,m} + f_v v_{l,m}\Big), \tag{3.2.50}$$

$$\frac{dv_{l,m}}{dt} = -d_\Omega k_{l,m}^2 v_{l,m} + \gamma_\Omega\Big(g_u u_{l,m} + g_v v_{l,m}\Big). \tag{3.2.51}$$

Equations (3.2.50) and (3.2.51) are supplemented with boundary conditions

$$u_{l,m}\psi'_{k_{l,m}}(1) = \gamma_\Gamma\Big(h_{1u}u_{l,m}\psi_{k_{l,m}}(1) + h_{1v}v_{l,m}\psi_{k_{l,m}}(1) + h_{1r}r_{l,m} + h_{1s}s_{l,m}\Big), \tag{3.2.52}$$

$$d_\Omega v_{l,m}\psi'_{k_{l,m}}(1) = \gamma_\Gamma\Big(h_{2u}u_{l,m}\psi_{k_{l,m}}(1) + h_{2v}v_{l,m}\psi_{k_{l,m}}(1) + h_{2r}r_{l,m} + h_{2s}s_{l,m}\Big), \tag{3.2.53}$$

where $\psi'_{k_{l,m}} := \frac{d\psi_{k_{l,m}}(r)}{dr}\Big|_{r=1}$. Writing

$$\Big(u_{l,m}, v_{l,m}, r_{l,m}, s_{l,m}\Big)^T = \Big(u^0_{l,m}, v^0_{l,m}, r^0_{l,m}, s^0_{l,m}\Big)^T e^{\lambda_{l,m}t},$$

and substituting into the system of ordinary differential equations (3.2.48)-(3.2.51), we obtain the following eigenvalue problem

$$\Big(\lambda_{l,m}\boldsymbol{I} + \boldsymbol{M}\Big)\boldsymbol{\xi}^0_{l,m} = \boldsymbol{0} \tag{3.2.54}$$

where

$$\boldsymbol{M} = \begin{pmatrix} k_{l,m}^2 - \gamma_\Omega f_u & -\gamma_\Omega f_v & 0 & 0 \\ -\gamma_\Omega g_u & d_\Omega k_{l,m}^2 - \gamma_\Omega g_v & 0 & 0 \\ \psi'_{k_{l,m}}(1) & 0 & l(l+1) - \gamma_\Gamma f_r & -\gamma_\Gamma f_s \\ 0 & d_\Omega\psi'_{k_{l,m}}(1) & -\gamma_\Gamma g_r & d_\Gamma l(l+1) - \gamma_\Gamma g_s \end{pmatrix},$$

and

$$\boldsymbol{\xi}^0_{l,m} = \Big(u^0_{l,m}, v^0_{l,m}, r^0_{l,m}, s^0_{l,m}\Big)^T.$$

Note that the boundary conditions (3.2.52) and (3.2.53) have been applied appropriately to the surface linearised reaction-diffusion equations. Since

$$\Big(u^0_{l,m}, v^0_{l,m}, r^0_{l,m}, s^0_{l,m}\Big)^T \neq \Big(0,0,0,0\Big)^T,$$

it follows that the coefficient matrix must be singular, hence we require that

$$\left|\lambda_{l,m}\boldsymbol{I}+\boldsymbol{M}\right|=0.$$

Straight forward calculations show that the eigenvalue $\lambda_{l,m}$ solves the following dispersion relation written in compact form as

$$\left(\lambda_{l,m}^2+\mathrm{Tr}\left(\boldsymbol{M}\right)_\Omega\lambda_{l,m}+\mathrm{Det}\left(\boldsymbol{M}\right)_\Omega\right)\left(\lambda_{l,m}^2+\mathrm{Tr}\left(\boldsymbol{M}\right)_\Gamma\lambda_{l,m}+\mathrm{Det}\left(\boldsymbol{M}\right)_\Gamma\right)=0, \quad (3.2.55)$$

where we have defined conveniently

$$\begin{aligned}
\mathrm{Tr}(\boldsymbol{M})_\Omega &:= (d_\Omega+1)k_{l,m}^2-\gamma_\Omega(f_u+g_v),\\
\mathrm{Tr}(\boldsymbol{M})_\Gamma &:= (d_\Gamma+1)l(l+1)-\gamma_\Gamma(f_r+g_s),\\
\mathrm{Det}(\boldsymbol{M})_\Omega &:= d_\Omega k_{l,m}^4-\gamma_\Omega\left(d_\Omega f_u+g_v\right)k_{l,m}^2+\gamma_\Omega^2(f_ug_v-f_vg_u),\\
\mathrm{Det}(\boldsymbol{M})_\Gamma &:= d_\Gamma l^2(l+1)^2-\gamma_\Gamma\left(d_\Gamma f_r+g_s\right)l(l+1)+\gamma_\Gamma^2(f_rg_s-f_sg_r).
\end{aligned}$$

The above holds true if and only if either

$$\lambda_{l,m}^2+\mathrm{Tr}\left(\boldsymbol{M}\right)_\Omega\lambda_{l,m}+\mathrm{Det}\left(\boldsymbol{M}\right)_\Omega=0, \quad (3.2.56)$$

or

$$\lambda_{l,m}^2+\mathrm{Tr}\left(\boldsymbol{M}\right)_\Gamma\lambda_{l,m}+\mathrm{Det}\left(\boldsymbol{M}\right)_\Gamma=0. \quad (3.2.57)$$

In the presence of diffusion, we require the emergence of spatial growth. In order for the uniform steady state $\boldsymbol{w}^*$ to be unstable we require that either

i. $\mathrm{Re}(\lambda_{l,m}(k_{l,m}^2))>0$ for some $k_{l,m}^2>0$,

   or

ii. $\mathrm{Re}(\lambda_{l,m}(l(l+1)))>0$ for some $l(l+1)>0$,

   or

iii. both.

Solving (3.2.56) (and similarly (3.2.57)) we obtain the eigenvalues

$$2\mathrm{Re}(\lambda_{l,m}(k_{l,m}^2))=-\mathrm{Tr}\left(\boldsymbol{M}\right)_\Omega\pm\sqrt{\mathrm{Tr}^2\left(\boldsymbol{M}\right)_\Omega-4\mathrm{Det}\left(\boldsymbol{M}\right)_\Omega}. \quad (3.2.58)$$

It follows then that $\mathrm{Re}(\lambda_{l,m}(k^2_{l,m})) > 0$ for some $k^2_{l,m} > 0$ if and only if the following conditions hold:

$$\begin{cases} \mathrm{Tr}\,(\boldsymbol{M})_\Omega < 0 \iff (d_\Omega + 1)k^2_{l,m} - \gamma_\Omega(f_u + g_v) < 0, \quad \text{and} \\ \\ \mathrm{Det}\,(\boldsymbol{M})_\Omega > 0 \iff d_\Omega k^4_{l,m} - \gamma_\Omega\,(d_\Omega f_u + g_v)\,k^2_{l,m} + \gamma^2_\Omega(f_u g_v - f_v g_u) > 0, \end{cases} \tag{3.2.59}$$

or

$$\begin{cases} \mathrm{Tr}\,(\boldsymbol{M})_\Omega > 0 \iff (d_\Omega + 1)k^2_{l,m} - \gamma_\Omega(f_u + g_v) > 0, \quad \text{and} \\ \\ \mathrm{Det}\,(\boldsymbol{M})_\Omega < 0 \iff d_\Omega k^4_{l,m} - \gamma_\Omega\,(d_\Omega f_u + g_v)\,k^2_{l,m} + \gamma^2_\Omega(f_u g_v - f_v g_u) < 0. \end{cases} \tag{3.2.60}$$

Similarly, on the surface, $\mathrm{Re}(\lambda_{l,m}(l(l+1))) > 0$ for some $l(l+1) > 0$ if and only the following conditions hold:

$$\begin{cases} \mathrm{Tr}\,(\boldsymbol{M})_\Gamma < 0 \iff (d_\Gamma + 1)l(l+1) - \gamma_\Gamma(f_r + g_s) < 0, \quad \text{and} \\ \\ \mathrm{Det}\,(\boldsymbol{M})_\Gamma > 0 \iff d_\Gamma l^2(l+1)^2 - \gamma_\Gamma\,(d_\Gamma f_r + g_s)\,l(l+1) + \gamma^2_\Gamma(f_r g_s - f_s g_r) > 0, \end{cases} \tag{3.2.61}$$

or

$$\begin{cases} \mathrm{Tr}\,(\boldsymbol{M})_\Gamma > 0 \iff (d_\Gamma + 1)l(l+1) - \gamma_\Gamma(f_r + g_s) > 0, \quad \text{and} \\ \\ \mathrm{Det}\,(\boldsymbol{M})_\Gamma < 0 \iff d_\Gamma l^2(l+1)^2 - \gamma_\Gamma\,(d_\Gamma f_r + g_s)\,l(l+1) + \gamma^2_\Gamma(f_r g_s - f_s g_r) < 0. \end{cases} \tag{3.2.62}$$

We are in a position to state the following theorem:

Theorem 3.2.2. *Assuming that*

$$Tr\left(\boldsymbol{J_F}\right)_\Omega = f_u + g_v < 0 \quad \textit{and} \quad Det\left(\boldsymbol{J_F}\right)_\Omega = f_u g_v - f_v g_u > 0, \tag{3.2.63}$$

then the necessary conditions for $Re(\lambda_{l,m}(k^2_{l,m})) > 0$ *for some* $k^2_{l,m} > 0$ *are given by*

$$d_\Omega f_u + g_v > 0, \quad \textit{and} \quad (d_\Omega f_u + g_v)^2 - 4d_\Omega\,(f_u g_v - f_v g_u) > 0. \tag{3.2.64}$$

Similarly, assuming that

$$Tr\left(\boldsymbol{J_F}\right)_\Gamma = f_r + g_s < 0 \quad \textit{and} \quad Det\left(\boldsymbol{J_F}\right)_\Gamma = f_r g_s - f_s g_r > 0, \tag{3.2.65}$$

then the necessary conditions for $Re(\lambda_{l,m}(l(l+1))) > 0$ *for some* $l(l+1) > 0$ *are given by*

$$d_\Gamma f_r + g_s > 0, \quad \textit{and} \quad (d_\Gamma f_r + g_s)^2 - 4d_\Gamma (f_r g_s - f_s g_r) > 0. \tag{3.2.66}$$

Proof. The proof is a direct consequence of conditions (3.2.59) - (3.2.62). Assuming conditions (3.2.63) and (3.2.65) hold, then one of the conditions in (3.2.59) and (3.2.61) is violated, which implies that $\mathrm{Re}(\lambda_{l,m}(k^2_{l,m})) < 0$ for all $k^2_{l,m} > 0$ and similarly $\mathrm{Re}(\lambda_{l,m}(l(l+1))) < 0$ for all $l(l+1) > 0$. This entails that the system can no longer exhibit spatially inhomogeneous solutions.

The only two conditions left to hold true are (3.2.60) and (3.2.62). This case corresponds to the classical standard two-component reaction-diffusion system which requires that (for details see for example (Murray 2003))

$$d_\Omega f_u + g_v > 0, \quad \text{and} \quad (d_\Omega f_u + g_v)^2 - 4d_\Omega (f_u g_v - f_v g_u) > 0, \tag{3.2.67}$$

and similarly

$$d_\Gamma f_r + g_s > 0, \quad \text{and} \quad (d_\Gamma f_r + g_s)^2 - 4d_\Gamma (f_r g_s - f_s g_r) > 0. \tag{3.2.68}$$

This completes the proof. □

Remark 3.2.3. Assuming conditions (3.2.63) and (3.2.65) both hold, then conditions (3.2.64) and (3.2.66) imply that $d_\Omega \neq 1$ and $d_\Gamma \neq 1$.

Remark 3.2.4. If condition (3.2.63) or (3.2.65) holds only, then either $d_\Omega \neq 1$ or $d_\Gamma \neq 1$ but not necessarily both.

Remark 3.2.5. If conditions (3.2.63) and (3.2.65) are both violated, then diffusion-driven instability can not occur.

Remark 3.2.6. Similar to classical reaction-diffusion systems, conditions (3.2.64) and (3.2.66) imply the existence of critical diffusion coefficients in the bulk and on the surface whereby the uniform states lose stability. In order for diffusion-driven instability to occur, the bulk and surface diffusion coefficients must be greater than the values of the critical diffusion coefficients.

Next we investigate under what assumptions on the reaction-kinetics do conditions (3.2.59) and (3.2.61) hold true.

- First let us consider the case when

$$\mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Omega = f_u + g_v > 0 \text{ and } \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Omega = f_u g_v - f_v g_u > 0,$$

and

$$\mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Gamma = f_r + g_s > 0 \text{ and } \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Gamma = f_r g_s - f_s g_r > 0.$$

Then $\mathrm{Tr}\left(\boldsymbol{J_F}\right) = \mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Omega + \mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Gamma > 0$ which violates condition (3.2.28).

- Similarly the case when

$$\mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Omega = f_u + g_v > 0 \text{ and } \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Omega = f_u g_v - f_v g_u < 0,$$

and

$$\mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Gamma = f_r + g_s > 0 \text{ and } \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Gamma = f_r g_s - f_s g_r < 0$$

violates condition (3.2.28).

- Let us consider the case when

$$\mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Omega = f_u + g_v < 0 \text{ and } \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Omega = f_u g_v - f_v g_u < 0,$$

and

$$\mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Gamma = f_r + g_s < 0 \text{ and } \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Gamma = f_r g_s - f_s g_r < 0.$$

Then it follows that condition (3.2.32) given by

$$\Big[\mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Gamma \mathrm{Tr}\left(\boldsymbol{J_F}\right) - 2\mathrm{Det}\left(\boldsymbol{J_F}\right)_\Omega\Big]\mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Omega + \Big[\mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Omega \mathrm{Tr}\left(\boldsymbol{J_F}\right) - 2\mathrm{Det}\left(\boldsymbol{J_F}\right)_\Gamma\Big]\mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Gamma < 0,$$

is violated.

- Next we consider the case when

$$\mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Omega = f_u + g_v < 0 \text{ and } \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Omega = f_u g_v - f_v g_u < 0,$$

and

$$\mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Gamma = f_r + g_s > 0 \text{ and } \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Gamma = f_r g_s - f_s g_r < 0.$$

It follows then that none of the conditions (3.2.28)-(3.2.33) are violated. However, condition (3.2.59) does not hold.

- Similarly the case when

$$\mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Omega = f_u + g_v > 0 \text{ and } \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Omega = f_u g_v - f_v g_u < 0,$$

and

$$\mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Gamma = f_r + g_s < 0 \text{ and } \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Gamma = f_r g_s - f_s g_r < 0.$$

This implies that that none of the conditions (3.2.28)-(3.2.33) are violated, while condition (3.2.61) fails not hold.

- Finally, the cases when

$$\begin{cases} \mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Omega = f_u + g_v > 0 \text{ and } \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Omega = f_u g_v - f_v g_u > 0, \\ \mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Gamma = f_r + g_s < 0 \text{ and } \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Gamma = f_r g_s - f_s g_r > 0, \end{cases} \tag{3.2.69}$$

and

$$\begin{cases} \mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Omega = f_u + g_v < 0 \text{ and } \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Omega = f_u g_v - f_v g_u > 0, \\ \mathrm{Tr}\left(\boldsymbol{J_F}\right)_\Gamma = f_r + g_s > 0 \text{ and } \mathrm{Det}\left(\boldsymbol{J_F}\right)_\Gamma = f_r g_s - f_s g_r > 0, \end{cases} \tag{3.2.70}$$

result in *Remark* 3.2.4 above.

The above cases clearly eliminate conditions (3.2.59) and (3.2.61) as necessary for uniform steady state to be driven unstable in the presence of diffusion. We are now in a position to state our main result.

Theorem 3.2.3 (Necessary conditions for diffusion-driven instability)**.** *The necessary conditions for diffusion-driven instability for the coupled system of BSRDEs* (3.2.1) - (3.2.4) *are given by*

$$Tr\left(\boldsymbol{J_F}\right) < 0, \tag{3.2.71}$$

$$Det\left(\boldsymbol{J_F}\right)_\Omega + Det\left(\boldsymbol{J_F}\right)_\Gamma + Tr\left(\boldsymbol{J_F}\right)_\Omega Tr\left(\boldsymbol{J_F}\right)_\Gamma > 0, \tag{3.2.72}$$

$$Det\left(\boldsymbol{J_F}\right)_\Omega Tr\left(\boldsymbol{J_F}\right)_\Gamma + Det\left(\boldsymbol{J_F}\right)_\Gamma Tr\left(\boldsymbol{J_F}\right)_\Omega < 0, \tag{3.2.73}$$

$$Det\left(\boldsymbol{J_F}\right)_\Omega Det\left(\boldsymbol{J_F}\right)_\Gamma > 0, \tag{3.2.74}$$

$$\left[Tr\left(\boldsymbol{J_F}\right)_\Gamma Tr\left(\boldsymbol{J_F}\right) - 2Det\left(\boldsymbol{J_F}\right)_\Omega\right] Tr\left(\boldsymbol{J_F}\right)_\Omega + \left[Tr\left(\boldsymbol{J_F}\right)_\Omega Tr\left(\boldsymbol{J_F}\right) - 2Det\left(\boldsymbol{J_F}\right)_\Gamma\right] Tr\left(\boldsymbol{J_F}\right)_\Gamma > 0, \tag{3.2.75}$$

$$\left[\left(Det\left(\boldsymbol{J_F}\right)_\Omega + Det\left(\boldsymbol{J_F}\right)_\Gamma\right)^2 - \left(Det\left(\boldsymbol{J_F}\right)_\Omega Tr\left(\boldsymbol{J_F}\right)_\Gamma + Det\left(\boldsymbol{J_F}\right)_\Gamma Tr\left(\boldsymbol{J_F}\right)_\Omega\right) Tr\left(\boldsymbol{J_F}\right)\right] Tr\left(\boldsymbol{J_F}\right)_\Omega Tr\left(\boldsymbol{J_F}\right)_\Gamma > 0, \tag{3.2.76}$$

and

$$d_\Omega f_u + g_v > 0, \quad \textit{and} \quad \left(d_\Omega f_u + g_v\right)^2 - 4 d_\Omega Det\left(\boldsymbol{J_F}\right)_\Omega > 0, \tag{3.2.77}$$

or/and

$$d_\Gamma f_r + g_s > 0, \quad \textit{and} \quad \left(d_\Gamma f_r + g_s\right)^2 - 4 d_\Gamma Det\left(\boldsymbol{J_F}\right)_\Gamma > 0. \tag{3.2.78}$$

Theoretical predictions

From the analytical results we state the following theoretical predictions to be validated through the use of numerical simulations.

i. The bulk and surface diffusion coefficients d_Ω and d_Γ must be greater than one in order for diffusion-driven instability to occur. Taking $d_\Omega = d_\Gamma = 1$ results in a contradiction between conditions (3.2.71), (3.2.77) and (3.2.78). As a result, the BSRDEs does not give rise to the formation of spatial structure. For this case, the uniform steady state is the only stable solution for the coupled system of BSRDEs (3.2.1) - (3.2.4).

ii. The above imply that taking $d_\Omega > 1$ and $d_\Gamma = 1$, the bulk-reaction-diffusion system has the potential to induce patterning in the bulk for appropriate diffusion-driven instability parameter values while the surface-reaction-diffusion system is not able to generate patterns. Here all the conditions (3.2.71)-(3.2.77) hold except (3.2.78).

iii. Alternatively taking $d_\Omega = 1$ and $d_\Gamma > 1$, the bulk-reaction-diffusion system fails to induce patterning in the bulk while the surface-reaction-diffusion system has the potential to induce patterning on the surface. Similarly, all the conditions (3.2.71)-(3.2.78) hold except (3.2.77).

iv. On the other hand, taking $d_\Omega > 1$ and $d_\Gamma > 1$ appropriately, then the coupled system of BSRDEs exhibits patterning both in the bulk and on the surface. All the conditions (3.2.71)-(3.2.78) hold both in the bulk and on the surface.

3.2.3 Weak variational form

Let $\varphi \in H^1(\Omega)$ and $\psi \in H^1(\Gamma)$. Then, multiplying (3.2.6) by φ and ψ, we seek $u, v \in L^2(0,t;H^1(\Omega))$ and $r, s \in L^2(0,t;H^1(\Gamma))$, such that $\forall \varphi \in H^1(\Omega)$ and $\forall \psi \in H^1(\Gamma)$

$$\begin{cases} \begin{cases} \displaystyle\int_\Omega (u_t\varphi + \nabla u \cdot \nabla \varphi) = \gamma_\Omega \int_\Omega (a - u + u^2 v)\varphi + \gamma_\Gamma \int_\Gamma (\alpha_1 r - \beta_1 u - \kappa_1 v)\varphi, \\ \displaystyle\int_\Omega (v_t\varphi + d_\Omega \nabla v \cdot \nabla \varphi) = \gamma_\Omega \int_\Omega (b - u^2 v)\varphi + \gamma_\Gamma \int_\Gamma (\alpha_2 s - \beta_2 u - \kappa_2 v)\varphi, \end{cases} \\ \qquad\qquad \boldsymbol{x} \text{ on } \Omega,\ t > 0, \\ \begin{cases} \displaystyle\int_\Gamma (r_t\psi + \nabla_\Gamma r \cdot \nabla_\Gamma \psi) = \gamma_\Gamma \int_\Gamma (a - r + r^2 s - \alpha_1 r + \beta_1 u + \kappa_1 v)\psi, \\ \displaystyle\int_\Gamma (s_t\psi + d_\Gamma \nabla_\Gamma s \cdot \nabla_\Gamma \psi) = \gamma_\Gamma \int_\Gamma (b - r^2 s - \alpha_2 s + \beta_2 u + \kappa_2 v)\psi, \end{cases} \\ \qquad\qquad \boldsymbol{x} \text{ on } \Gamma,\ t > 0. \end{cases} \tag{3.2.79}$$

In the above, we have used Green's identities with the boundary conditions (3.2.2) to obtain the boundary integrals.

3.2.4 Discretisation

In this section we present the bulk-surface finite element discretisation corresponding to the coupled system of bulk-surface reaction diffusion equations (BSRDEs) (3.2.1)-(3.2.5). Our method is inspired by the work of Elliott & Ranner (2012) and extends the discretisation in the previous chapter. We use the bulk-surface finite element method to discretise in space with piecewise bilinear elements and an implicit second order fractional-step θ-scheme to discretise in time using Newton's method for the lineraisation (Madzvamuse & Chung 2014*b*,*a*). For details on the convergence and stability of the fully implicit time-stepping fractional-step θ-scheme, the reader is referred to Madzvamuse *et al.* (Madzvamuse & Chung 2014*b*,*a*).

Let $\Omega_h \subset \Omega$ and $\Gamma_h \subset \Gamma$ be discretisations of the original domains with N_Ω and N_Γ vertices respectively. Let us take the finite dimensional subspaces $V_{\Omega_h} \subset H^1(\Omega)$ and $V_{\Gamma_h} \subset H^1(\Gamma)$. Then, we seek $u_h, v_h \in L^2(0,t;V_{\Omega_h})$ and $r_h, s_h \in L^2(0,t;V_{\Gamma_h})$ such that for

all $\varphi_h \in V_{\Omega_h}$ and $\psi_h \in V_{\Gamma_h}$

$$\begin{cases} \begin{cases} \displaystyle\int_{\Omega_h} (u_{h,t}\varphi_h + \nabla u_h \cdot \nabla \varphi_h) = \gamma_\Omega \int_{\Omega_h} (a - u_h + u_h^2 v_h)\varphi_h + \gamma_\Gamma \int_{\Gamma_h} (\alpha_1 r_h - \beta_1 u_h - \kappa_1 v_h)\varphi_h, \\ \qquad\qquad \boldsymbol{x} \text{ on } \Omega_h,\ t > 0, \\ \displaystyle\int_{\Omega_h} (v_{h,t}\varphi_h + d_\Omega \nabla v_h \cdot \nabla \varphi_h) = \gamma_\Omega \int_{\Omega_h} (b - u_h^2 v_h)\varphi_h + \gamma_\Gamma \int_{\Gamma_h} (\alpha_2 s_h - \beta_2 u_h - \kappa_2 v_h)\varphi_h, \end{cases} \\ \begin{cases} \displaystyle\int_{\Gamma_h} (r_{h,t}\psi_h + \nabla_\Gamma r_h \cdot \nabla_\Gamma \psi_h) = \gamma_\Gamma \int_{\Gamma_h} (a - r_h + r_h^2 s_h - \alpha_1 r_h + \beta_1 u_h + \kappa_1 v_h)\psi_h, \\ \qquad\qquad \boldsymbol{x} \text{ on } \Gamma_h,\ t > 0. \\ \displaystyle\int_{\Gamma_h} (s_{h,t}\psi_h + d_\Gamma \nabla_\Gamma s_h \cdot \nabla_\Gamma \psi_h) = \gamma_\Gamma \int_{\Gamma_h} (b - r_h^2 s_h - \alpha_2 s_h + \beta_2 u_h + \kappa_2 v_h)\psi_h\,. \end{cases} \end{cases}$$

Let $\{\varphi_i\}_{i=1}^{N_\Omega}$ and $\{\psi_i\}_{i=1}^{N_\Gamma}$ be the set of piecewise bilinear shape functions on Ω_h and Γ_h respectively. Then the sets form a basis of V_Ω and V_Γ. Thus, we can write the discrete solution variables as $u_h = \Sigma_{i=1}^{N_\Omega} u_i \varphi_i := \boldsymbol{u}\cdot\boldsymbol{\varphi}$, and similarly for the other variables. Then, the above can be written compactly in matrix form as

$$\begin{cases} \begin{cases} M_{\boldsymbol{\varphi}}\boldsymbol{u}_t + \gamma_\Omega M_{\boldsymbol{\varphi}}\boldsymbol{u} + A_{\boldsymbol{\varphi}}\boldsymbol{u} - \gamma_\Omega B_{\boldsymbol{\varphi}}(\boldsymbol{u},\boldsymbol{v})\boldsymbol{u} \\ \qquad -\gamma_\Gamma\big(\alpha_1 M_{\varphi\psi}\boldsymbol{r} - \beta_1 M_{\varphi\varphi}\boldsymbol{u} - \kappa_1 M_{\varphi\varphi}\boldsymbol{v}\big) = \gamma_\Omega a \mathbb{1}_{\boldsymbol{\varphi}}, \\ M_{\boldsymbol{\varphi}}\boldsymbol{v}_t + d_\Omega A_{\boldsymbol{\varphi}}\boldsymbol{v} + \gamma_\Omega B_{\boldsymbol{\varphi}}(\boldsymbol{u},\boldsymbol{u})\boldsymbol{v} \\ \qquad -\gamma_\Gamma\big(\alpha_2 M_{\varphi\psi}\boldsymbol{s} - \beta_2 M_{\varphi\varphi}\boldsymbol{u} - \kappa_2 M_{\varphi\varphi}\boldsymbol{v}\big) = \gamma_\Omega b \mathbb{1}_{\boldsymbol{\varphi}}, \end{cases} \quad \text{in } \Omega_h \times (0,T], \\ \begin{cases} M_{\boldsymbol{\psi}}\boldsymbol{r}_t + \gamma_\Gamma M_{\boldsymbol{\psi}}\boldsymbol{r} + A_{\boldsymbol{\psi}}\boldsymbol{r} - \gamma_\Gamma B_{\boldsymbol{\psi}}(\boldsymbol{r},\boldsymbol{s})\boldsymbol{r} \\ \qquad +\gamma_\Gamma\big(\alpha_1 M_{\psi\psi}\boldsymbol{r} - \beta_1 M_{\psi\varphi}\boldsymbol{u} - \kappa_1 M_{\psi\varphi}\boldsymbol{v}\big) = \gamma_\Gamma a \mathbb{1}_{\boldsymbol{\psi}}, \\ M_{\boldsymbol{\psi}}\boldsymbol{s}_t + d_\Gamma A_{\boldsymbol{\psi}}\boldsymbol{s} + \gamma_\Gamma B_{\boldsymbol{\psi}}(\boldsymbol{r},\boldsymbol{r})\boldsymbol{s} \\ \qquad +\gamma_\Gamma\big(\alpha_2 M_{\psi\psi}\boldsymbol{s} - \beta_2 M_{\psi\varphi}\boldsymbol{u} - \kappa_2 M_{\psi\varphi}\boldsymbol{v}\big) = \gamma_\Gamma b \mathbb{1}_{\boldsymbol{\psi}}, \end{cases} \quad \text{on } \Gamma_h \times (0,T], \end{cases} \tag{3.2.80}$$

where $A_{\boldsymbol{\varphi}}$ and $M_{\boldsymbol{\varphi}}$ are the stiffness and mass matrices respectively with entries

$$(A_{\boldsymbol{\varphi}})_{ij} = \int_{\Omega_h} \nabla\varphi_i \cdot \nabla\varphi_j \, d\boldsymbol{x} \quad \text{and} \quad (M_{\boldsymbol{\varphi}})_{ij} = \int_{\Omega_h} \varphi_i\varphi_j \, d\boldsymbol{x}. \tag{3.2.81}$$

$\mathbb{1}_\varphi$ is the column vector with j-th entry $\int_{\Omega_h} \varphi_j$. Given some vectors $\boldsymbol{a}$, $\boldsymbol{b}$ and $\boldsymbol{c}$, $B_{\boldsymbol{\varphi}}(\boldsymbol{a},\boldsymbol{b})$ is the matrix with entries

$$(B_{\boldsymbol{\varphi}})_{ij} = \int_{\Omega_h} (\boldsymbol{a}\cdot\boldsymbol{\varphi})(\boldsymbol{b}\cdot\boldsymbol{\varphi})\varphi_i\varphi_j \, d\boldsymbol{x}. \tag{3.2.82}$$

Similar quantities are define for $A_{\boldsymbol{\psi}}$, $M_{\boldsymbol{\psi}}$, $\mathbb{1}_{\psi}$ and $B_{\boldsymbol{\psi}}$. The matrices $M_{\varphi\varphi}$, $M_{\varphi\psi}$, $M_{\psi\varphi}$ and $M_{\psi\psi}$ have entries

$$(M_{\varphi\varphi})_{ij} = \int_{\Gamma_h} \varphi_i\varphi_j d\boldsymbol{x},\ (M_{\varphi\psi})_{ij} = \int_{\Gamma_h} \varphi_i\psi_j d\boldsymbol{x},\ (M_{\psi\varphi})_{ij} = \int_{\Gamma_h} \psi_i\varphi_j d\boldsymbol{x},\ (M_{\psi\psi})_{ij} = \int_{\Gamma_h} \psi_i\psi_j d\boldsymbol{x}.$$

After the investigation in the previous chapter, we choose to discretise in time using the fractional-step θ method and treat the non-linearities using the Newton method (see Section 2.2.4). Thus, let T_m denote the maximum time of interest, τ denote the time step and J be a fixed nonnegative integer, then

$$\tau = \frac{T_m}{J} \quad \text{and} \quad t_k = k\tau, \quad k = 0, 1, 2, \ldots, J.$$

We denote the approximate solution at time t_k by $u_h^k = u_h(\cdot, t_k) = \boldsymbol{u}^k \cdot \varphi$ and similarly for the other variables. Following from (3.2.80), the fractional-step θ method is implemented as follows:

Starting with the previous solution $(\boldsymbol{u}^n, \boldsymbol{v}^n, \boldsymbol{r}^n, \boldsymbol{s}^n)$, we first solve for the intermediate solution $(\boldsymbol{u}^{n+\theta}, \boldsymbol{v}^{n+\theta}, \boldsymbol{r}^{n+\theta}, \boldsymbol{s}^{n+\theta})$

$$\begin{cases}
M_{\boldsymbol{\varphi}} \dfrac{\boldsymbol{u}^{n+\theta} - \boldsymbol{u}^n}{\theta\tau} + A_{\boldsymbol{\varphi}}\boldsymbol{u}^{n+\theta} + \gamma_\Omega M_{\boldsymbol{\varphi}}\boldsymbol{u}^{n+\theta} \\
- \gamma_\Gamma\big(\alpha_1 M_{\varphi\psi}\boldsymbol{r}^{n+\theta} - \beta_1 M_{\varphi\varphi}\boldsymbol{u}^{n+\theta} - \kappa_1 M_{\varphi\varphi}\boldsymbol{v}^{n+\theta}\big) = \gamma_\Omega a\mathbb{1}_\varphi + \gamma_\Omega B_{\boldsymbol{\varphi}}(\boldsymbol{u}^n, \boldsymbol{v}^n)\boldsymbol{u}^n, \\
M_{\boldsymbol{\varphi}} \dfrac{\boldsymbol{v}^{n+\theta} - \boldsymbol{v}^n}{\theta\tau} + dA_{\boldsymbol{\varphi}}\boldsymbol{v}^{n+\theta} \\
- \gamma_\Gamma\big(\alpha_2 M_{\varphi\psi}\boldsymbol{s}^{n+\theta} - \beta_2 M_{\varphi\varphi}\boldsymbol{u}^{n+\theta} - \kappa_2 M_{\varphi\varphi}\boldsymbol{v}^{n+\theta}\big) = \gamma_\Omega b\mathbb{1}_\varphi - \gamma_\Omega B_{\boldsymbol{\varphi}}(\boldsymbol{u}^n, \boldsymbol{u}^n)\boldsymbol{v}^n, \\
M_{\boldsymbol{\psi}} \dfrac{\boldsymbol{r}^{n+\theta} - \boldsymbol{r}^n}{\theta\tau} + A_{\boldsymbol{\psi}}\boldsymbol{r}^{n+\theta} + \gamma_\Gamma M_{\boldsymbol{\psi}}\boldsymbol{r}^{n+\theta} \\
+ \gamma_\Gamma\big(\alpha_1 M_{\psi\psi}\boldsymbol{r}^{n+\theta} - \beta_1 M_{\psi\varphi}\boldsymbol{u}^{n+\theta} - \kappa_1 M_{\psi\varphi}\boldsymbol{v}^{n+\theta}\big) = \gamma_\Gamma a\mathbb{1}_\psi + \gamma_\Gamma B_{\boldsymbol{\psi}}(\boldsymbol{r}^n, \boldsymbol{s}^n)\boldsymbol{r}^n, \\
M_{\boldsymbol{\psi}} \dfrac{\boldsymbol{s}^{n+\theta} - \boldsymbol{s}^n}{\theta\tau} + dA_{\boldsymbol{\psi}}\boldsymbol{s}^{n+\theta} \\
+ \gamma_\Gamma\big(\alpha_2 M_{\psi\psi}\boldsymbol{s}^{n+\theta} - \beta_2 M_{\psi\varphi}\boldsymbol{u}^{n+\theta} - \kappa_2 M_{\psi\varphi}\boldsymbol{v}^{n+\theta}\big) = \gamma_\Gamma b\mathbb{1}_\psi - \gamma_\Gamma B_{\boldsymbol{\psi}}(\boldsymbol{r}^n, \boldsymbol{r}^n)\boldsymbol{s}^n,
\end{cases}$$

then for the second intermediate solution $(\boldsymbol{u}^{n+1-\theta}, \boldsymbol{v}^{n+1-\theta}, \boldsymbol{r}^{n+1-\theta}, \boldsymbol{s}^{n+1-\theta})$ we solve

$$\begin{cases} M_{\boldsymbol{\varphi}} \dfrac{\boldsymbol{u}^{n+1-\theta} - \boldsymbol{u}^{n+\theta}}{(1-2\theta)\tau} - \gamma_\Omega B_{\boldsymbol{\varphi}}(\boldsymbol{u}^{n+1-\theta}, \boldsymbol{v}^{n+1-\theta})\boldsymbol{u}^{n+1-\theta} = \gamma_\Omega a \mathbb{1}_\varphi \\ - A_{\boldsymbol{\varphi}} \boldsymbol{u}^{n+\theta} - \gamma_\Omega M_{\boldsymbol{\varphi}} \boldsymbol{u}^{n+\theta} + \gamma_\Gamma \big(\alpha_1 M_{\varphi\psi} \boldsymbol{r}^{n+\theta} - \beta_1 M_{\varphi\varphi} \boldsymbol{u}^{n+\theta} - \kappa_1 M_{\varphi\varphi} \boldsymbol{v}^{n+\theta}\big) \\ M_{\boldsymbol{\varphi}} \dfrac{\boldsymbol{v}^{n+1-\theta} - \boldsymbol{v}^{n+\theta}}{(1-2\theta)\tau} + \gamma_\Omega B_{\boldsymbol{\varphi}}(\boldsymbol{u}^{n+1-\theta}, \boldsymbol{u}^{n+1-\theta})\boldsymbol{v}^{n+1-\theta} = \gamma_\Omega b \mathbb{1}_\varphi \\ - d A_{\boldsymbol{\varphi}} \boldsymbol{v}^{n+\theta} + \gamma_\Gamma \big(\alpha_2 M_{\varphi\psi} \boldsymbol{s}^{n+\theta} - \beta_2 M_{\varphi\varphi} \boldsymbol{u}^{n+\theta} - \kappa_2 M_{\varphi\varphi} \boldsymbol{v}^{n+\theta}\big), \\ M_{\boldsymbol{\psi}} \dfrac{\boldsymbol{r}^{n+1-\theta} - \boldsymbol{r}^{n+\theta}}{(1-2\theta)\tau} - \gamma_\Gamma B_{\boldsymbol{\psi}}(\boldsymbol{r}^{n+1-\theta}, \boldsymbol{s}^{n+1-\theta})\boldsymbol{r}^{n+1-\theta} = \gamma_\Gamma a \mathbb{1}_\psi \\ - A_{\boldsymbol{\psi}} \boldsymbol{r}^{n+\theta} - \gamma_\Gamma M_{\boldsymbol{\psi}} \boldsymbol{r}^{n+\theta} - \gamma_\Gamma \big(\alpha_1 M_{\psi\psi} \boldsymbol{r}^{n+\theta} - \beta_1 M_{\psi\varphi} \boldsymbol{u}^{n+\theta} - \kappa_1 M_{\psi\varphi} \boldsymbol{v}^{n+\theta}\big), \\ M_{\boldsymbol{\psi}} \dfrac{\boldsymbol{s}^{n+1-\theta} - \boldsymbol{s}^{n+\theta}}{(1-2\theta)\tau} + \gamma_\Gamma B_{\boldsymbol{\psi}}(\boldsymbol{r}^{n+1-\theta}, \boldsymbol{r}^{n+1-\theta})\boldsymbol{s}^{n+1-\theta} = \gamma_\Gamma b \mathbb{1}_\psi \\ - d A_{\boldsymbol{\psi}} \boldsymbol{s}^{n+\theta} - \gamma_\Gamma \big(\alpha_2 M_{\psi\psi} \boldsymbol{s}^{n+\theta} - \beta_2 M_{\psi\varphi} \boldsymbol{u}^{n+\theta} - \kappa_2 M_{\psi\varphi} \boldsymbol{v}^{n+\theta}\big), \end{cases} \tag{3.2.83}$$

and finally for the new solution $(\boldsymbol{u}^{n+1}, \boldsymbol{v}^{n+1}, \boldsymbol{r}^{n+1}, \boldsymbol{s}^{n+1})$ we solve

$$\begin{cases} M_{\boldsymbol{\varphi}} \dfrac{\boldsymbol{u}^{n+1} - \boldsymbol{u}^{n+1-\theta}}{\theta\tau} + A_{\boldsymbol{\varphi}} \boldsymbol{u}^{n+1} + \gamma_\Omega M_{\boldsymbol{\varphi}} \boldsymbol{u}^{n+1} \\ - \gamma_\Gamma \big(\alpha_1 M_{\varphi\psi} \boldsymbol{r}^{n+1} - \beta_1 M_{\varphi\varphi} \boldsymbol{u}^{n+1} - \kappa_1 M_{\varphi\varphi} \boldsymbol{v}^{n+1}\big) = \gamma_\Omega a \mathbb{1}_\varphi + \gamma_\Omega B_{\boldsymbol{\varphi}}(\boldsymbol{u}^{n+1-\theta}, \boldsymbol{v}^{n+1-\theta})\boldsymbol{u}^{n+1-\theta}, \\ M_{\boldsymbol{\varphi}} \dfrac{\boldsymbol{v}^{n+1} - \boldsymbol{v}^{n+1-\theta}}{\theta\tau} + d A_{\boldsymbol{\varphi}} \boldsymbol{v}^{n+1} \\ - \gamma_\Gamma \big(\alpha_2 M_{\varphi\psi} \boldsymbol{s}^{n+1} - \beta_2 M_{\varphi\varphi} \boldsymbol{u}^{n+1} - \kappa_2 M_{\varphi\varphi} \boldsymbol{v}^{n+1}\big) = \gamma_\Omega b \mathbb{1}_\varphi - \gamma_\Omega B_{\boldsymbol{\varphi}}(\boldsymbol{u}^{n+1-\theta}, \boldsymbol{u}^{n+1-\theta})\boldsymbol{v}^{n+1-\theta}, \\ M_{\boldsymbol{\psi}} \dfrac{\boldsymbol{r}^{n+1} - \boldsymbol{r}^{n+1-\theta}}{\theta\tau} + A_{\boldsymbol{\psi}} \boldsymbol{r}^{n+1} + \gamma_\Gamma M_{\boldsymbol{\psi}} \boldsymbol{r}^{n+1} \\ + \gamma_\Gamma \big(\alpha_1 M_{\psi\psi} \boldsymbol{r}^{n+1} - \beta_1 M_{\psi\varphi} \boldsymbol{u}^{n+1} - \kappa_1 M_{\psi\varphi} \boldsymbol{v}^{n+1}\big) = \gamma_\Gamma a \mathbb{1}_\psi + \gamma_\Gamma B_{\boldsymbol{\psi}}(\boldsymbol{r}^{n+1-\theta}, \boldsymbol{s}^{n+1-\theta})\boldsymbol{r}^{n+1-\theta}, \\ M_{\boldsymbol{\psi}} \dfrac{\boldsymbol{s}^{n+1} - \boldsymbol{s}^{n+1-\theta}}{\theta\tau} + d A_{\boldsymbol{\psi}} \boldsymbol{s}^{n+1} \\ + \gamma_\Gamma \big(\alpha_2 M_{\psi\psi} \boldsymbol{s}^{n+1} - \beta_2 M_{\psi\varphi} \boldsymbol{u}^{n+1} - \kappa_2 M_{\psi\varphi} \boldsymbol{v}^{n+1}\big) = \gamma_\Gamma b \mathbb{1}_\psi - \gamma_\Gamma B_{\boldsymbol{\psi}}(\boldsymbol{r}^{n+1-\theta}, \boldsymbol{r}^{n+1-\theta})\boldsymbol{s}^{n+1-\theta}. \end{cases}$$

3.3 Computation Implementation

The computation implementation here makes use of the code described in Section 2.3. In the coupled system some terms can be obtained from the original code, but other terms will need to be newly calculated. For instance, in the first step of the fractional-step θ method, the matrix system can be rewritten

$$\big(M_1 + M_2\big) \begin{pmatrix} \boldsymbol{u}^{n+\theta} \\ \boldsymbol{v}^{n+\theta} \\ \boldsymbol{r}^{n+\theta} \\ \boldsymbol{s}^{n+\theta} \end{pmatrix} = \begin{pmatrix} \frac{1}{\theta\tau} M_{\boldsymbol{\varphi}} \boldsymbol{u}^n + \gamma_\Omega a \mathbb{1}_\varphi + \gamma_\Omega B_{\boldsymbol{\varphi}}(\boldsymbol{u}^n, \boldsymbol{v}^n)\boldsymbol{u}^n \\ \frac{1}{\theta\tau} M_{\boldsymbol{\varphi}} \boldsymbol{v}^n + \gamma_\Omega b \mathbb{1}_\varphi - \gamma_\Omega B_{\boldsymbol{\varphi}}(\boldsymbol{u}^n, \boldsymbol{u}^n)\boldsymbol{v}^n \\ \frac{1}{\theta\tau} M_{\boldsymbol{\psi}} \boldsymbol{r}^n + \gamma_\Gamma a \mathbb{1}_\psi + \gamma_\Gamma B_{\boldsymbol{\psi}}(\boldsymbol{r}^n, \boldsymbol{s}^n)\boldsymbol{r}^n, \\ \frac{1}{\theta\tau} M_{\boldsymbol{\psi}} \boldsymbol{s}^n + \gamma_\Gamma b \mathbb{1}_\psi - \gamma_\Gamma B_{\boldsymbol{\psi}}(\boldsymbol{r}^n, \boldsymbol{r}^n)\boldsymbol{s}^n, \end{pmatrix} \tag{3.3.1}$$

where

$$M_1 = \begin{pmatrix} \left(\frac{1}{\theta\tau} + \gamma_\Omega\right) M_{\boldsymbol{\varphi}} + A_{\boldsymbol{\varphi}} & 0 & 0 & 0 \\ 0 & \frac{1}{\theta\tau} M_{\boldsymbol{\varphi}} + d_\Omega A_{\boldsymbol{\varphi}} & 0 & 0 \\ 0 & 0 & \left(\frac{1}{\theta\tau} + \gamma_\Gamma\right) M_{\boldsymbol{\psi}} + A_{\boldsymbol{\psi}} & 0 \\ 0 & 0 & 0 & \frac{1}{\theta\tau} M_{\boldsymbol{\psi}} + d_\Omega A_{\boldsymbol{\psi}} \end{pmatrix} \tag{3.3.2}$$

and

$$M_2 = \begin{pmatrix} \gamma_\Gamma \beta_1 M_{\varphi\varphi} & \gamma_\Gamma \kappa_1 & -\gamma_\Gamma \alpha_1 M_{\varphi\psi} & 0 \\ \gamma_\Gamma \beta_2 M_{\varphi\varphi} & \gamma_\Gamma \kappa_2 M_{\varphi\varphi} & 0 & -\gamma_\Gamma \alpha_2 M_{\varphi\psi} \\ -\gamma_\Gamma \beta_1 M_{\psi\varphi} & -\gamma_\Gamma \kappa_1 M_{\psi\varphi} & \gamma_\Gamma \alpha_1 M_{\psi\psi} & 0 \\ -\gamma_\Gamma \beta_2 M_{\psi\varphi} & -\gamma_\Gamma \kappa_2 M_{\psi\varphi} & 0 & \gamma_\Gamma \alpha_2 M_{\psi\psi} \end{pmatrix}. \tag{3.3.3}$$

It is seen that the original code can provide the terms in M_1 and provide the matrices $B_{\boldsymbol{\varphi}}(\boldsymbol{u}^n, \boldsymbol{v}^n)$ and $B_{\boldsymbol{\psi}}(\boldsymbol{r}^n, \boldsymbol{s}^n)$ at each time-step; the new terms appear in the matrix M_2. Similar points applies to the other sub-steps in the fractional-step θ method. For the computational implementation, we use a structure that can hold the system matrix of the entire coupled system, make use of the original code to get the relevant terms and also assemble the new additional terms. This is all collected in a new class

```
template <int spacedim>
class CoupledSystemHandler
{
 public:
  CoupledSystemHandler (Parameters &parameters);
  void run ();
 private:
  SchnakProblem<spacedim,spacedim>   bulk_system;
  SchnakProblem<spacedim,spacedim-1> surface_system;
  Matrix                             full_system_LHS_matrix;
  Vector                             full_system_RHS_vector;
  ...
  void                               assemble_system();
};
```

This class has two instances of the class *SchnakProblem*, one for the bulk and one for the surface geometry. The class is a template class which takes the space dimension as its only parameter. We ensure that surface_system contains a surface geometry by passing

through the relevant template parameters. The class has a run() function which allows the user to carry out the simulation without seeing the details of the implementation. The class also contains a matrix and vector structure to hold the full system of equations.

The structure of the run() function remains largely the same as in Section 2.3. What is perhaps most important is the way in which the assembly is done

```
template <int spacedim>
void CoupledSystemHandler<spacedim>::assemble_system()
{
 // get contributions from bulk_system
 // get contributions from surface_system
 // get additional contirubtions
 // add all contibutions to full_system_LHS_matrix and full_system_RHS_vector
}
```

This process is made to run in parallel - one thread can call the assemble() function of bulk_system whilst another can call the assemble() function of surface_system at the same time. Once these are obtained, the additional contributions can also be obtained in parallel as described in Section 2.3. Finally, all contributions are added to the whole system and subsequently solved.

3.4 Numerical simulations of the coupled system of bulk-surface reaction-diffusion equations (BSRDEs)

In all our numerical experiments, we fix the kinetic model parameter values $a = 0.1$ and $b = 0.9$ since these satisfy the Turing diffusion-driven instability conditions (3.2.71)-(3.2.78). For these model parameter values the equilibrium values are $(u^*, v^*, r^*, s^*) = (1, 0.9, 1, 0.9)$. Initial conditions are prescribed as small random perturbations around the equilibrium values. For illustrative purposes let us take the parameter values describing the boundary conditions as shown in Table 3.1; these are selected such that they satisfy the compatibility condition (3.2.14). We present numerical simulations of the coupled

a	b	γ_Ω	γ_Γ	α_1	α_2	β_1	β_2	κ_1	κ_2
0.1	0.9	500	500	$\frac{5}{12}$	5	$\frac{5}{12}$	0	0	5

Table 3.1: Model parameter values for the coupled system of BSRDEs (3.2.1) - (3.2.4).

system of BSRDEs on three different volumes: a sphere of radius one, a cube of length one and a triaxial ellipsoid with semi-axis of lengths 1, 2 an 3. In all our simulations we the parameters in Table 3.1 and vary the diffusion coefficients as shown in Table 3.2. The

Figure	d_Ω	d_Γ
3.3	1.0	1.0
3.4	1.0	10.0
3.5	1.0	20.0
3.6	10.0	1.0
3.7	10.0	10.0
3.8	20.0	10.0
3.9	20.0	20.0

Table 3.2: Model parameter values used in simulations for Figures 3.3-3.9.

.

numbers of degrees of freedom for the the sphere were 59042/3076 (bulk/surface), for the cube 71874/12292 and for the ellipsoid 59042/3076. The timestep chosen in each case was $\tau = 10^{-3}$. The results are shown in Figures 3.3-3.9.

Our numerical simulations reveal the following fundamental characteristics key to the theory of pattern formation:

i. By taking $d_\Omega = 1$ in the bulk and $d_\Gamma = 1$ on the surface, no patterns emerge (see Figure 3.3). This implies that the system fails to satisfy one of the necessary conditions for the formation of spatial structure which requires that one of the molecular species must diffuse much faster (typically the inhibitor) than the other (typically the activator), resulting in what is known as the *long-range inhibition short-range activation* (Gierer & Meinhardt 1972, Murray 2003). This entails that both the bulk and surface dynamics are not able to generate patterns, instead the uniform steady state is the only stable solution.

ii. Taking $d_\Omega = 1$ in the bulk and $d_\Gamma \gg 1$ on the surface, the bulk reaction-diffusion system is not able to generate patterns everywhere in the bulk. However, the surface reaction-diffusion system is able to form patterns on the surface as well as inducing patterning in a small band (which we can be considered as an epidermis) inside the bulk, close to the surface (see Figures 3.4 and 3.5).

iii. On the other hand, taking $d_\Omega \gg 1$ in the bulk and $d_\Gamma = 1$ on the surface, the bulk reaction-diffusion system is now able to generate patterns everywhere in the bulk including on the surface. The surface reaction-diffusion system is not able to generate patterns as expected, instead a uniform state is obtained (see Figure 3.6).

iv. If we take $d_\Omega \gg 1$ in the bulk and $d_\Gamma \gg 1$ on the surface, then the coupled bulk-surface reaction-diffusion system generates patterns in the interior as well as on the surface (see Figures 3.7, 3.8 and 3.9).

The results obtained are independent of the geometry; the sphere, cube and ellipsoid volumes all produced similar results. Furthermore, our results hold true for different values of the diffusion coefficients d_Ω in the bulk and d_Γ on the surface (further results not shown), and the scaling parameters γ_Ω and γ_Γ (further results not shown). Taking large values of γ_Ω and γ_Γ results in more complex patterns forming while for the same values of the diffusion coefficients as detailed above, similar patterning behaviour is observed.

A note on the bulk-surface triangulation

We briefly outline how the bulk-surface triangulation is generated. Let Ω_h be a triangulation of the bulk geometry Ω with vertices $\boldsymbol{x}_i$, $i = 1, ..., N_h$, where N_h is the number of vertices in the triangulation. From Ω_h we then construct Γ_h to be the triangulation of the surface geometry Γ by defining $\Gamma_h = \Omega_h|_{\partial\Omega_h}$, i.e. the vertices of Γ_h are the same as those lying on the surface of Ω_h. In particular, then, we have $\partial\Omega_h = \Gamma_h$. An example mesh is shown in Figure 3.2. The bulk triangulation induces the surface triangulation as illustrated.

3.4.1 Investigating the boundary behaviour

From Fig. 3.4-3.6 one can see an apparent large gap between the values on the surface and the values in the bulk. For the parameters $d_\Omega = 1$ and $d_\Gamma > 1$ in the simulations, the interior of the bulk is at the steady state, whilst the surface of the bulk has patterns which correspond to the surface solutions. For the parameters $d_\Omega > 1$ and $d_\Gamma = 1$ in our simulations, the interior of the bulk solution has patterns and the surface of the bulk solution is at the steady state, but the surface solution has patterns.

In order to get a closer look at the transition of the bulk solution from the surface to the interior, line plots of the solution u for the sphere are plotted in Fig. 3.10. These were

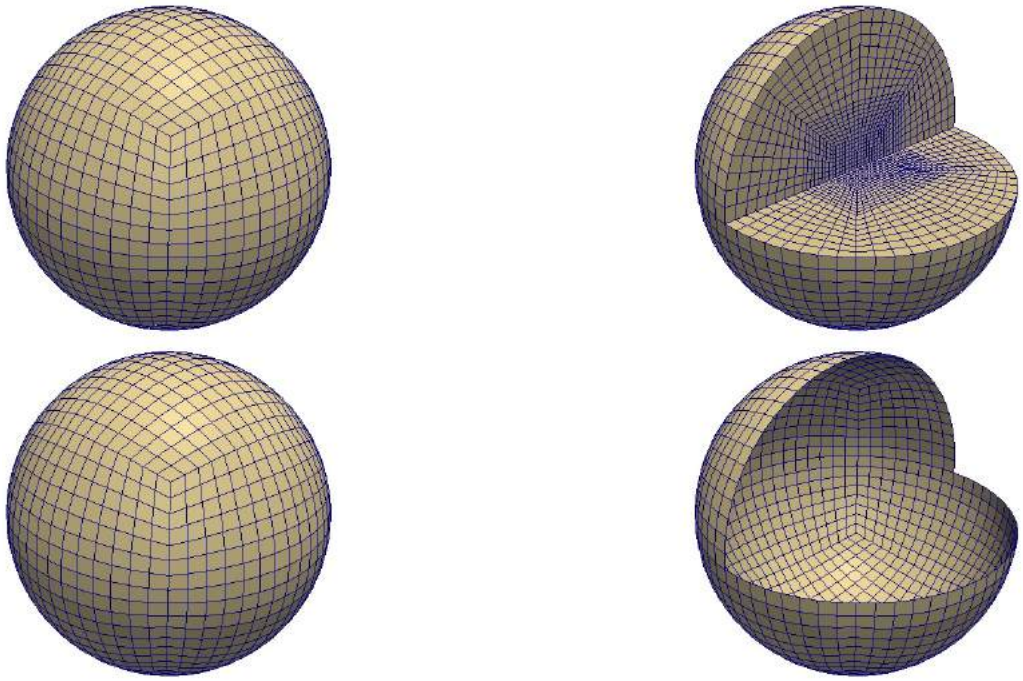


Figure 3.2: Example meshes for the bulk (top) and surface system (bottom). Part of the domain has been cut away and shown on the right to reveal some internal mesh structure.

obtained by plotting the solution u along the straight line from the centre of the sphere (the origin) to a point on the surface of the boundary. From these plots we make the following observations:

- Fig. 3.10(a) shows the solution u for $d_\Omega = 1$ and varying values of d_Γ. This graph corresponds to the solutions shown in Fig. 3.3-3.5. Here the transition from the boundary value to the steady state in the interior can be seen clearly. For larger values of d_Γ, the value on the boundary is higher and thus there is a greater difference between the boundary value and the steady state in the interior. With the values of d_Γ used, the value of u seems to reach the steady state value in the interior at the same distance from the origin.

- Fig. 3.10(b) shows the solution u for $d_\Gamma = 1$ and varying values of d_Γ. The line graphs show patterning in the solution, where greater values of d_Ω results in greater amplitudes in the patterning. The value on the boundary is fixed at the steady state value.

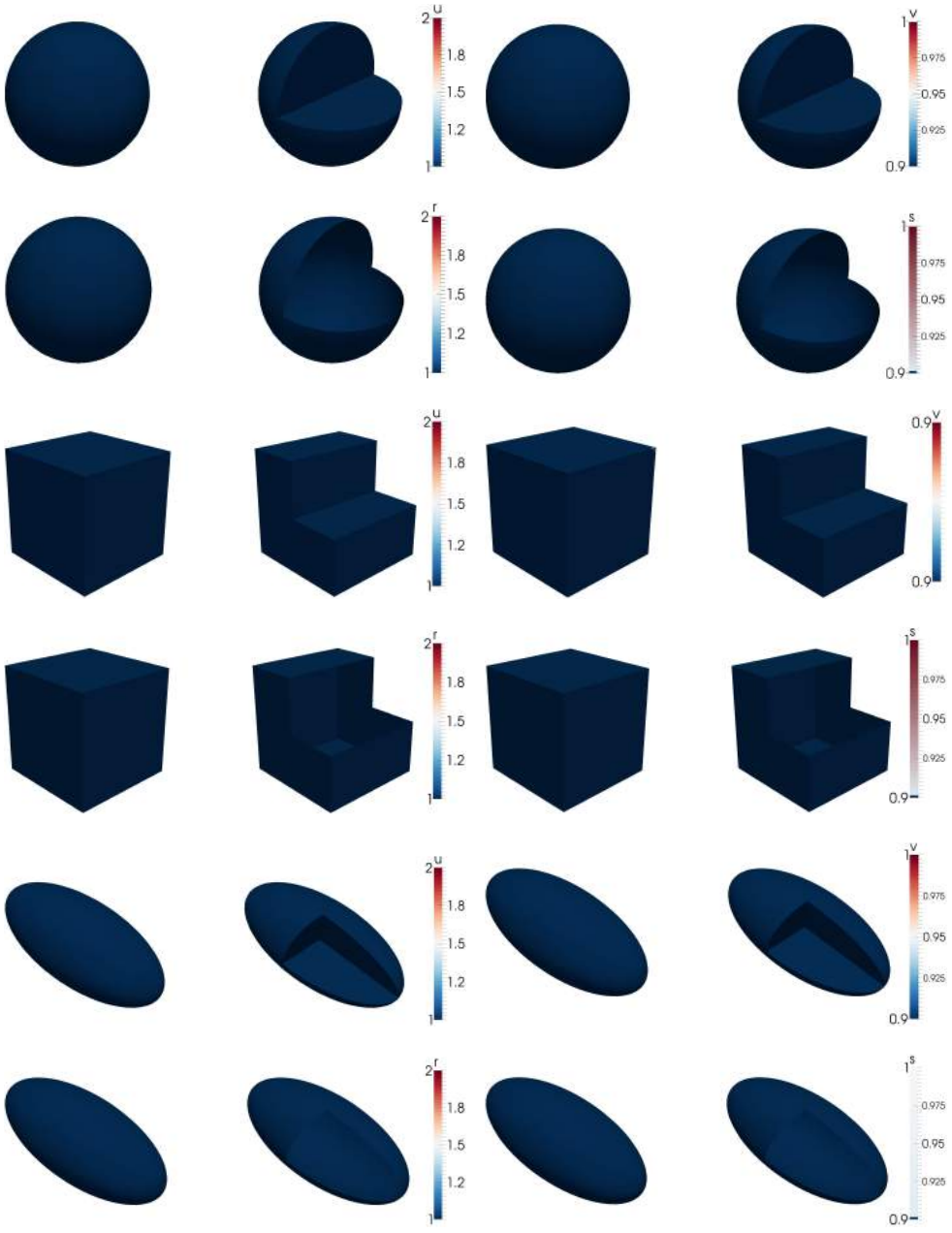


Figure 3.3: Numerical solutions corresponding to the coupled system of BSRDEs (3.2.1)-(3.2.5) with $d_\Omega = 1$ in the bulk, $d_\Gamma = 1$ on the surface and $\gamma_\Omega = \gamma_\Gamma = 500$. The uniform steady state solutions are converged to and no patterns form. Rows 1, 3, and 5: solutions in the bulk representing u and v. Rows 2, 4 and 6: solutions on the surface representing r and s. Second and fourth columns represent cross sections of the bulk and the surface respectively.

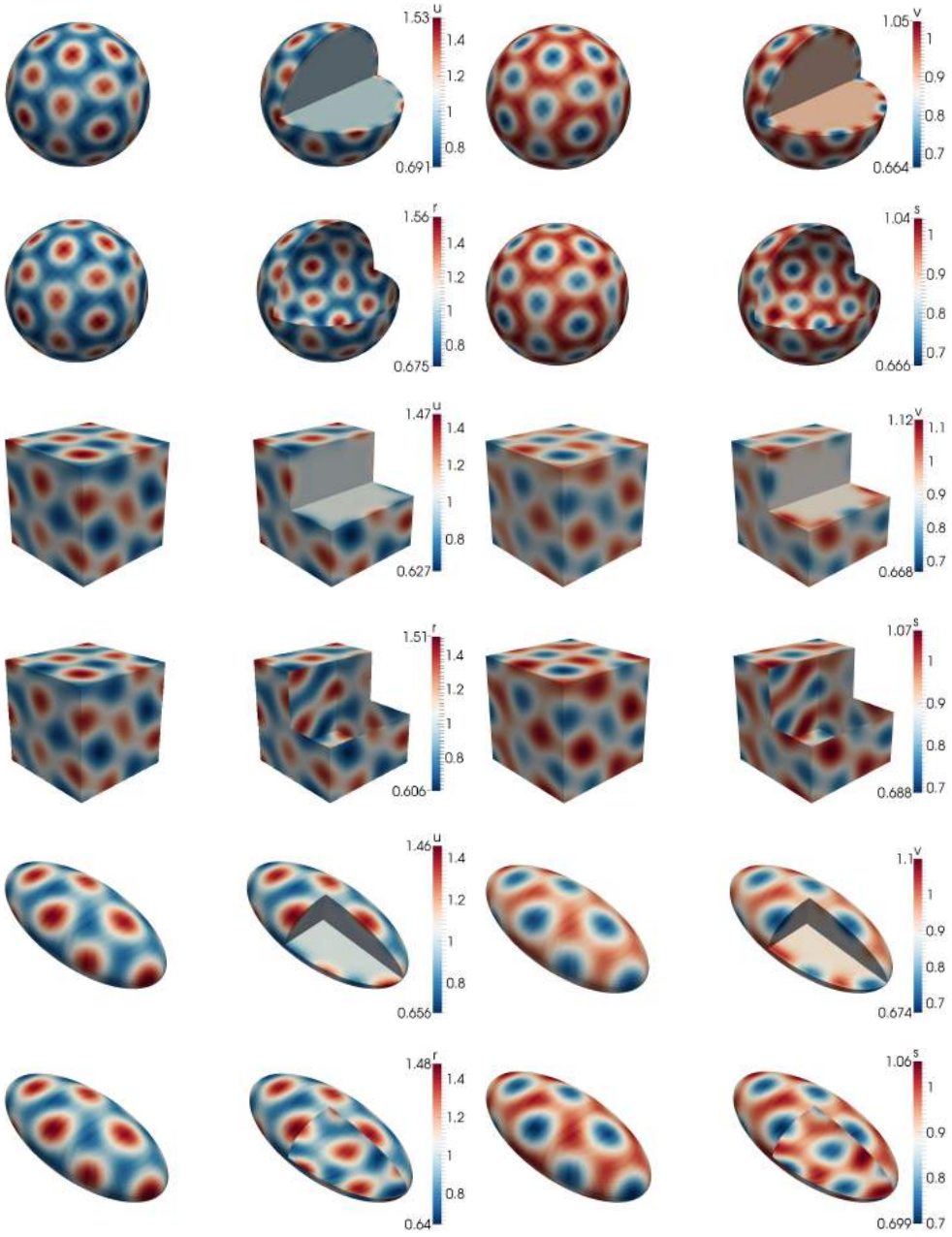


Figure 3.4: Numerical solutions corresponding to the coupled system of BSRDEs (3.2.1)-(3.2.5) with $d_\Omega = 1$ in the bulk, $d_\Gamma = 10$ on the surface and $\gamma_\Omega = \gamma_\Gamma = 500$. The surface reaction-diffusion system induces patterning in a small band close to the surface. In the bulk, no patterns form almost everywhere. The patterning behaviour is independent of the geometry. Rows 1, 3, and 5: solutions in the bulk representing u and v. Rows 2, 4 and 6: solutions on the surface representing r and s. Second and fourth columns represent cross sections of the bulk and the surface respectively.

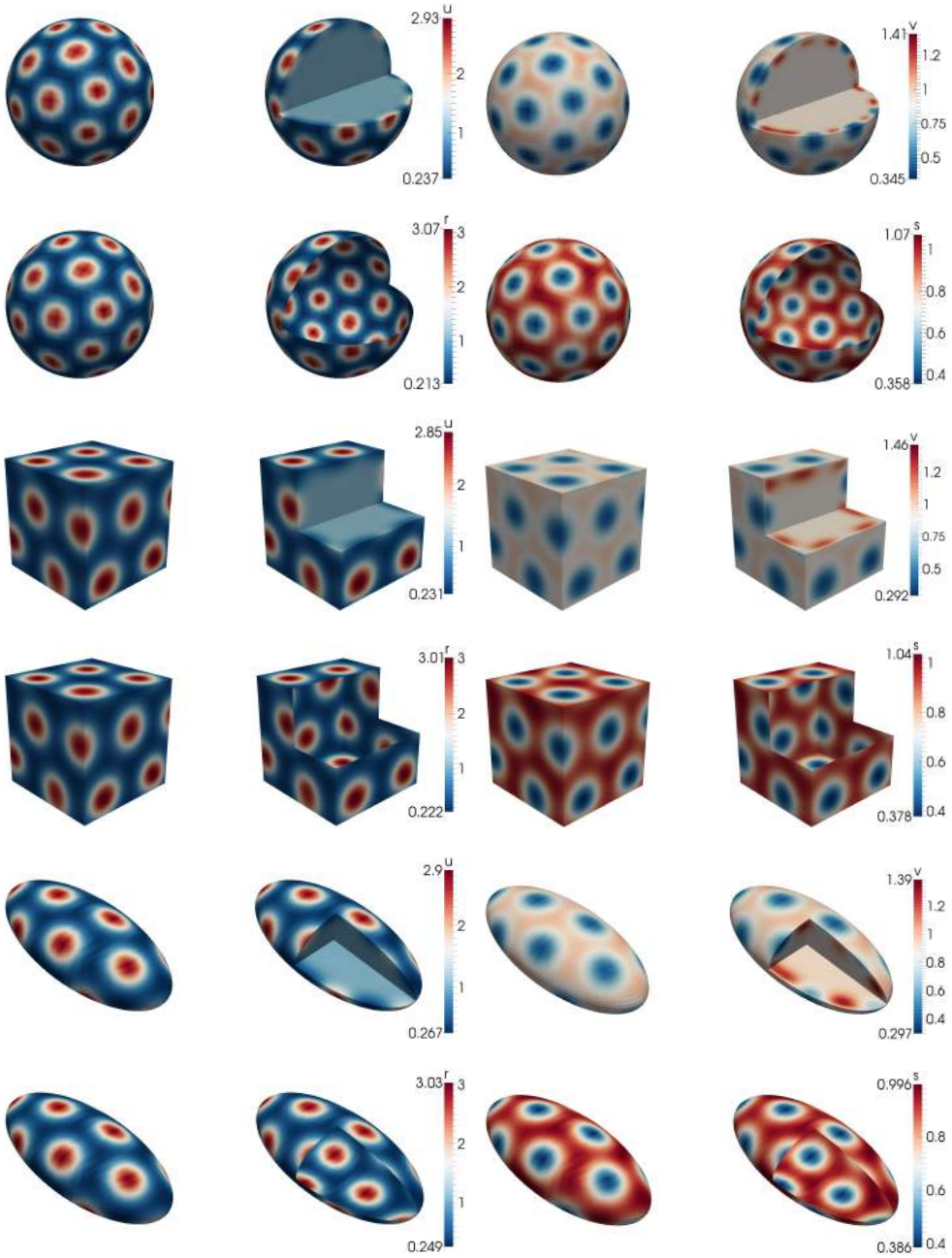


Figure 3.5: Numerical solutions corresponding to the coupled system of BSRDEs (3.2.1)-(3.2.5) with $d_\Omega = 1$ in the bulk, $d_\Gamma = 20$ on the surface and $\gamma_\Omega = \gamma_\Gamma = 500$. The pattering process is similar to that shown in Figure 3.4 for large values of d_Γ. Rows 2, 4 and 6: solutions on the surface representing r and s. Second and fourth columns represent cross sections of the bulk and the surface respectively.

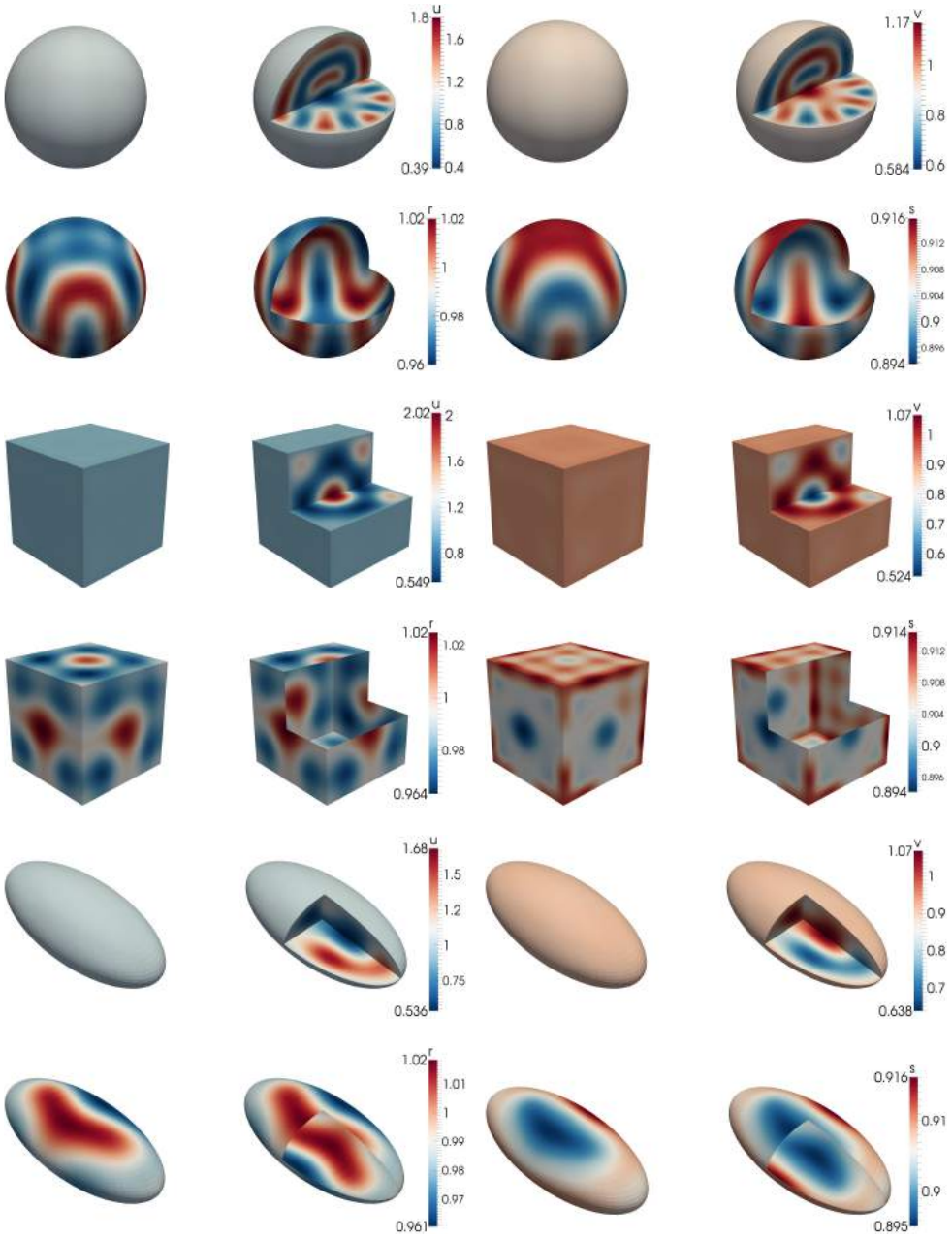


Figure 3.6: Numerical solutions corresponding to the coupled system of BSRDEs (3.2.1)-(3.2.5) with $d_\Omega = 10$ in the bulk, $d_\Gamma = 1$ on the surface and $\gamma_\Omega = \gamma_\Gamma = 500$. The bulk reaction-diffusion system is able to induce patterning almost everywhere on the surface. Rows 2, 4 and 6: solutions on the surface representing r and s. Second and fourth columns represent cross sections of the bulk and the surface respectively.

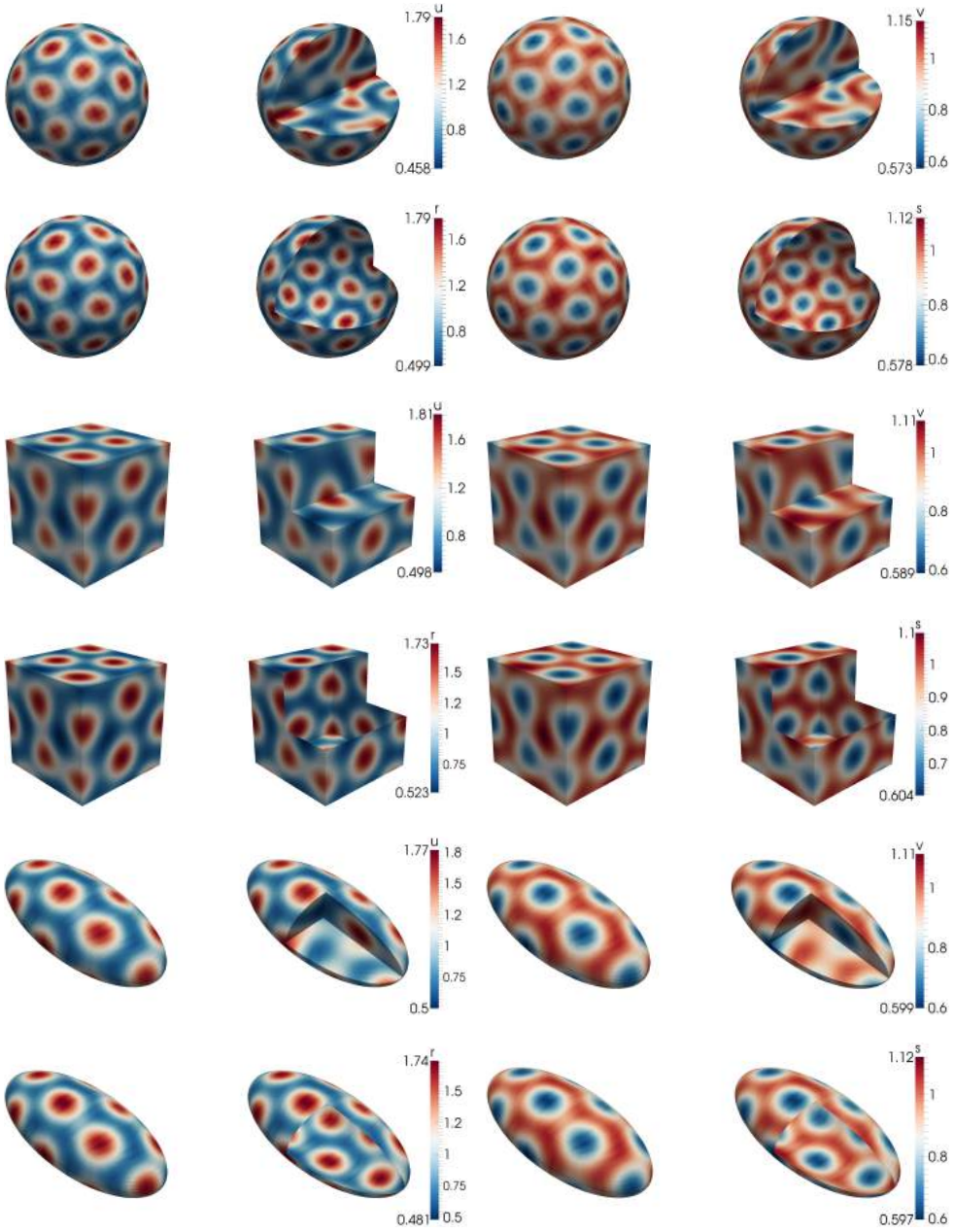


Figure 3.7: Numerical solutions corresponding to the coupled system of BSRDEs (3.2.1)-(3.2.5) with $d_\Omega = 10$ in the bulk, $d_\Gamma = 10$ on the surface and $\gamma_\Omega = \gamma_\Gamma = 500$. The bulk and surface reaction-diffusion systems both have the capability to induce patterning. Rows 2, 4 and 6: solutions on the surface representing r and s. Second and fourth columns represent cross sections of the bulk and the surface respectively.

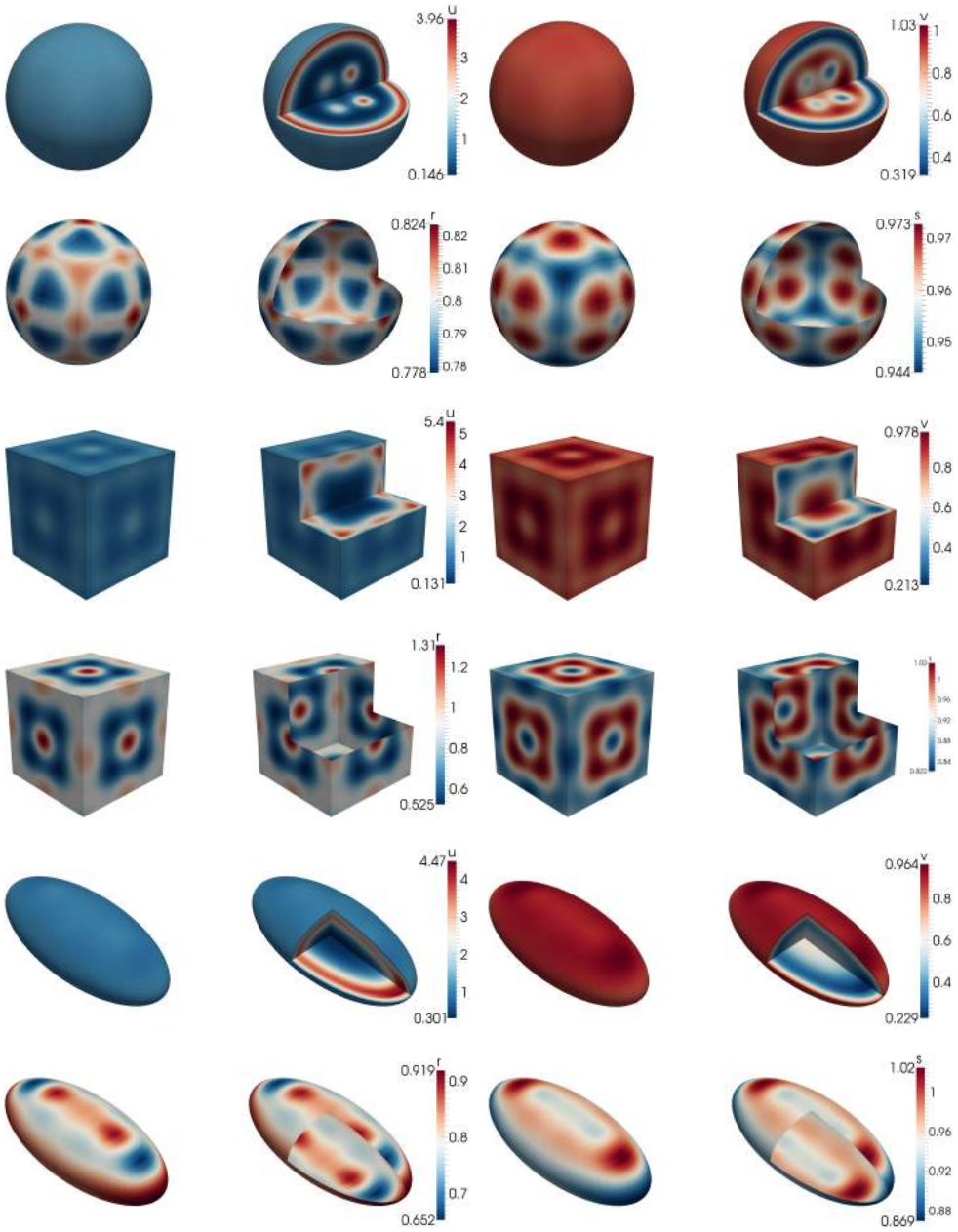


Figure 3.8: Numerical solutions corresponding to the coupled system of BSRDEs (3.2.1)-(3.2.5) with $d_\Omega = 20$ in the bulk, $d_\Gamma = 10$ on the surface and $\gamma_\Omega = \gamma_\Gamma = 500$. The coupled system of BSRDEs induces pattering in the bulk and on the surface. Rows 2, 4 and 6: solutions on the surface representing r and s. Second and fourth columns represent cross sections of the bulk and the surface respectively.

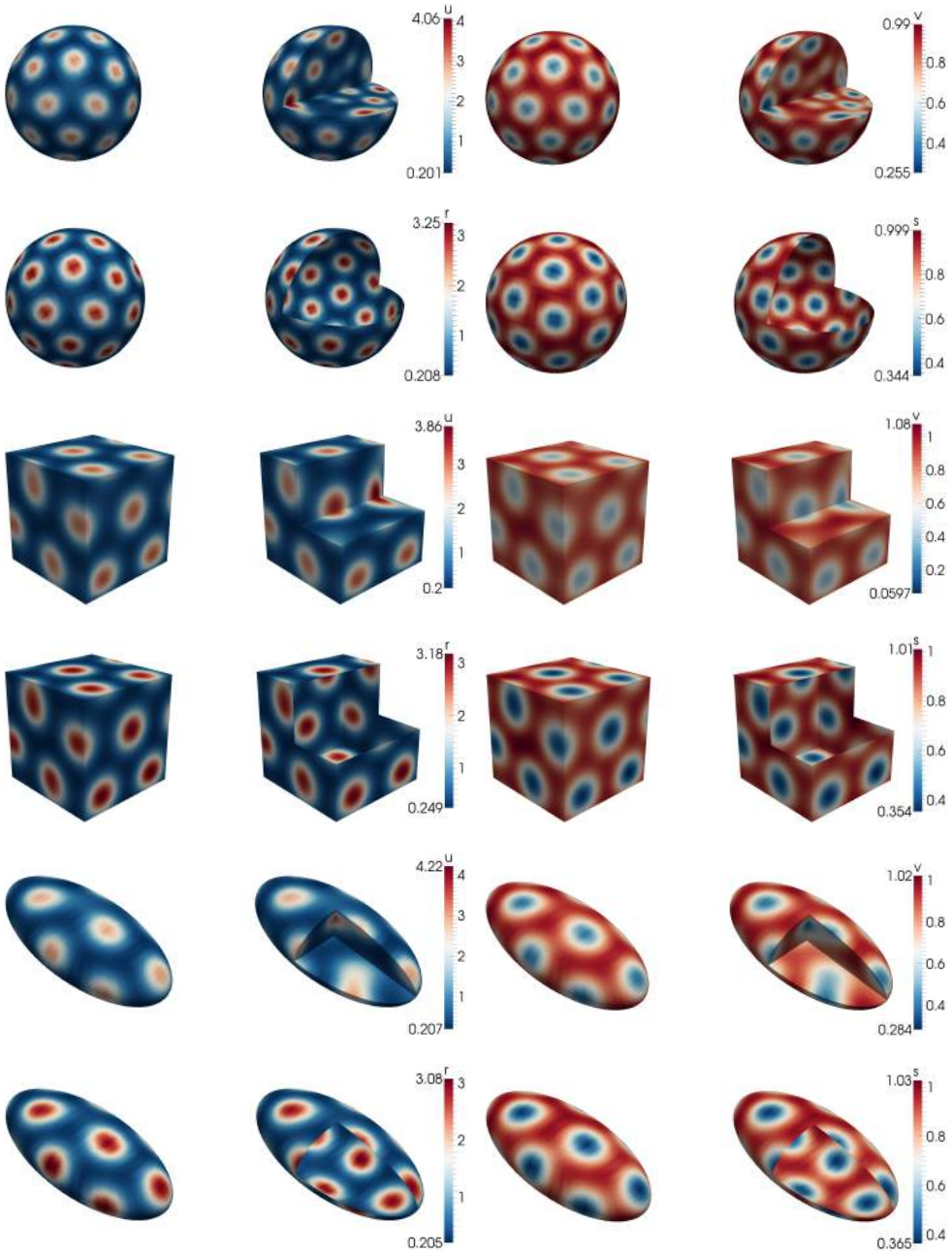
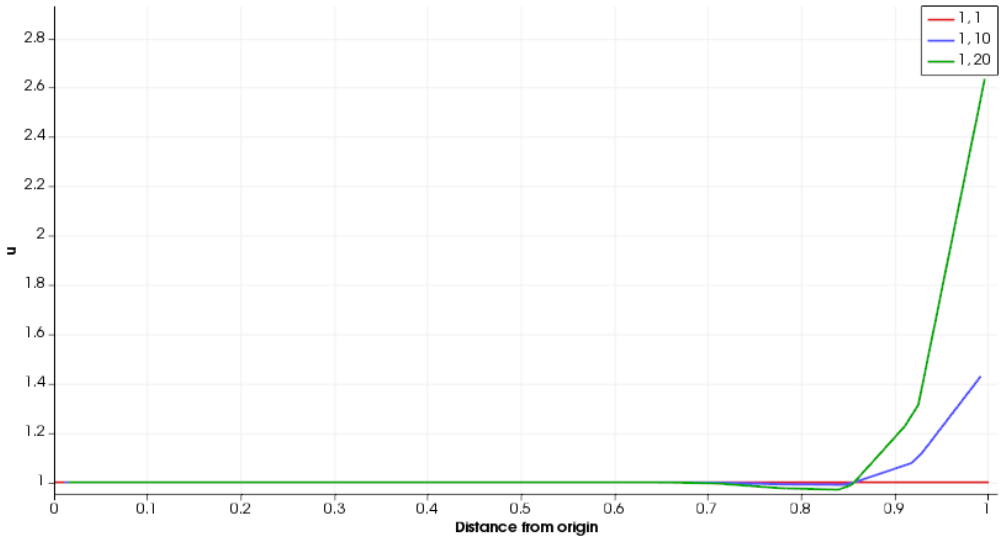
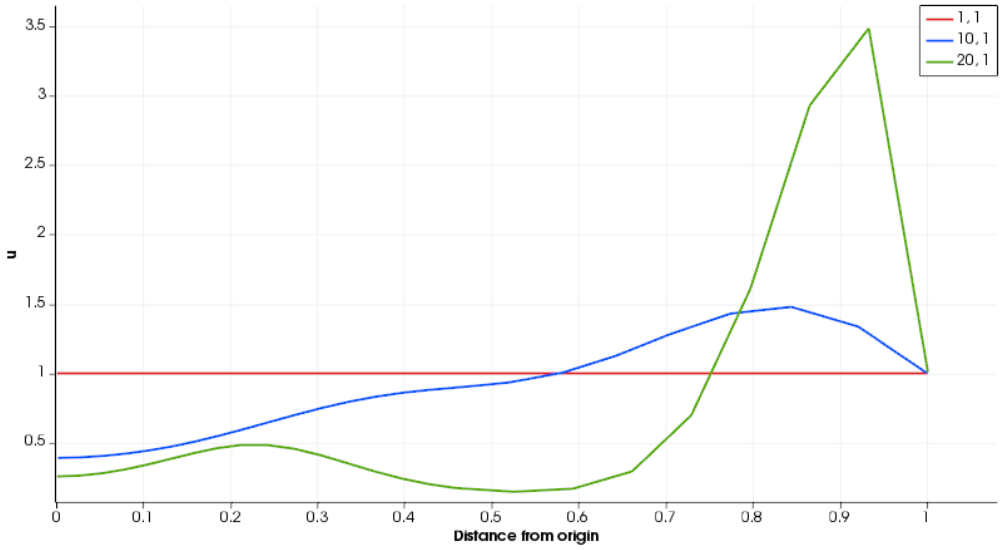


Figure 3.9: Numerical solutions corresponding to the coupled system of BSRDEs (3.2.1)-(3.2.5) with $d_\Omega = 20$ in the bulk, $d_\Gamma = 20$ on the surface and $\gamma_\Omega = \gamma_\Gamma = 500$. Patterning behaviour is more complex for large values of the diffusion coefficients. Rows 2, 4 and 6: solutions on the surface representing r and s. Second and fourth columns represent cross sections of the bulk and the surface, respectively.



(a) $d_\Omega = 1$



(b) $d_\Gamma = 1$

Figure 3.10: Line plots of the solution u for the simulations in the unit sphere centred on the origin with varying values of d_Ω and d_Γ. The solution is plotted along the straight line extending from the origin to: (a) a point on the sphere with the maximum boundary value of u; (b) the point $(1, 0, 0)$. The two values in the legends correspond to the values (d_Ω, d_Γ) in that order. The top graph holds the value $d_\Omega = 1$ fixed whilst the bottom graph holds the value $d_\Gamma = 1$ fixed. The value of 0 on the horizontal axis corresponds to the origin and the value of 1 corresponds to a point on the surface of the spherical domain.

3.5 Weak vs. Strong Coupling

Let us investigate the effect of increasing the coupling parameters. Let us take parameter values as shown in Table 3.3. It is easily verified that these choice of parameters satisfy

a	b	γ_Ω	γ_Γ	d_Ω	d_Γ	α_1	α_2	β_1	β_2	κ_1	κ_2
0.1	0.9	10	10	10	10	$2\varepsilon b$	ε	εb	$\frac{\varepsilon b}{2}$	ε	$\frac{\varepsilon}{2}$

Table 3.3: Model parameter values for the coupled system of BSRDEs (3.2.1) - (3.2.4), where ε is some non-negative parameter.

the compatibility conditions (3.2.12). Choosing the parameters this way, the parameter ε represents the relative magnitude of the coupling parameters. The case $\varepsilon = 0$ represents no coupling and is equivalent to taking homogeneous Neumann boundary conditions. We calculate the numerical solution using a number of values for ε shown in Table 3.4. The

Figure	ε
3.11	0
3.12	0.001
3.13	0.01
3.14	0.1
3.15	1
3.11	10

Table 3.4: Model parameter values used in simulations for Figures 3.11-3.16

.

result presented in Figures 3.11-3.16 show that for weaker coupling the magnitude of the solutions is closer to the case where there is no coupling. Conversely, as the coupling gets stronger, the magnitude of the solutions become increasingly different to the case with no coupling.

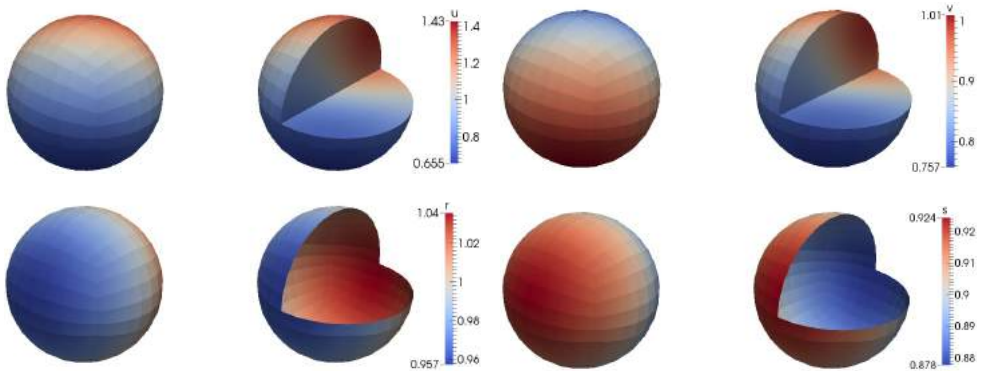


Figure 3.11: Numerical solution to the coupled system of BSRDEs (3.2.1) - (3.2.4) using the parameter values shown in Table 3.3 with $\varepsilon = 0$

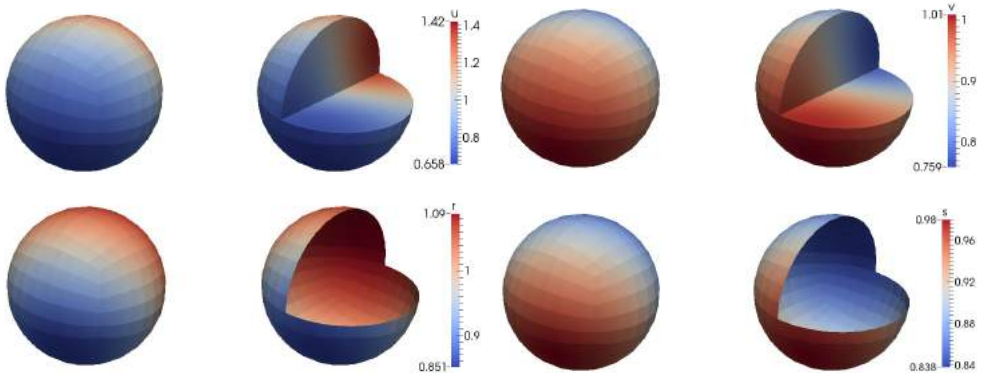


Figure 3.12: Numerical solution to the coupled system of BSRDEs (3.2.1) - (3.2.4) using the parameter values shown in Table 3.3 with $\varepsilon = 0.001$

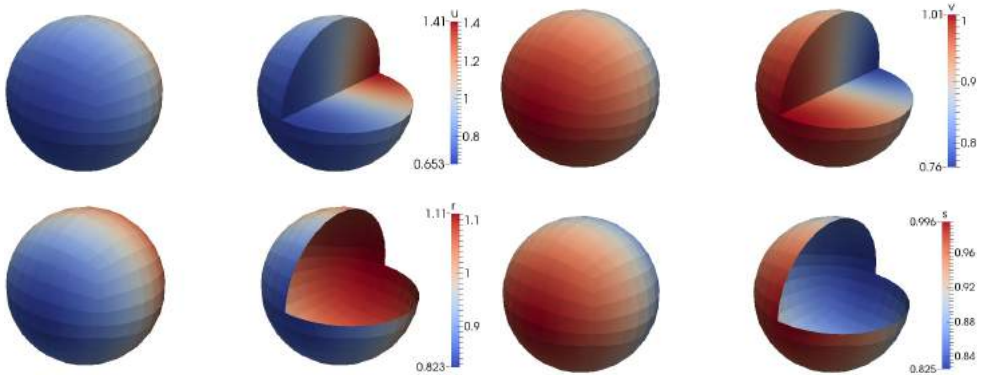


Figure 3.13: Numerical solution to the coupled system of BSRDEs (3.2.1) - (3.2.4) using the parameter values shown in Table 3.3 with $\varepsilon = 0.01$

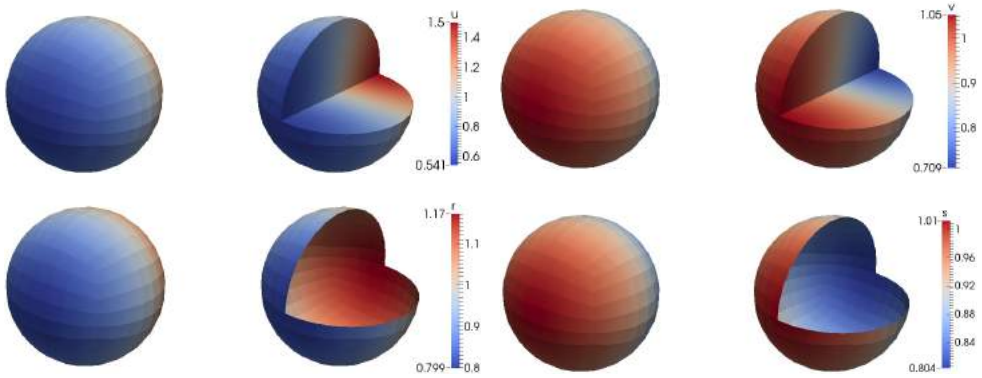


Figure 3.14: Numerical solution to the coupled system of BSRDEs (3.2.1) - (3.2.4) using the parameter values shown in Table 3.3 with $\varepsilon = 0.1$

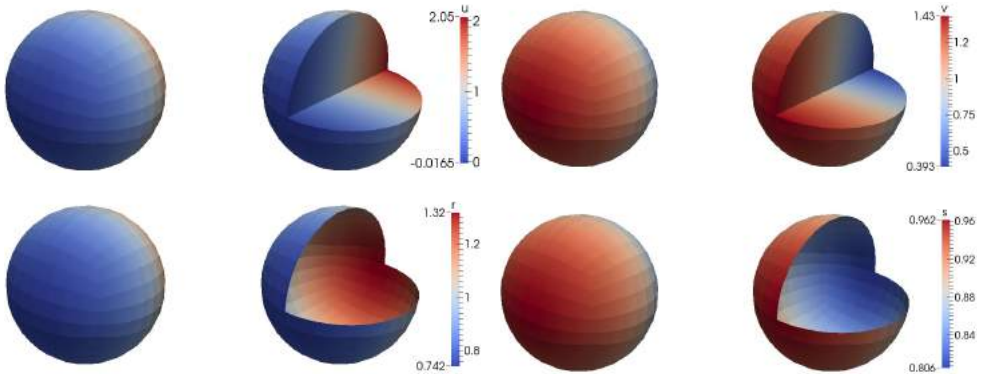


Figure 3.15: Numerical solution to the coupled system of BSRDEs (3.2.1) - (3.2.4) using the parameter values shown in Table 3.3 with $\varepsilon = 1$

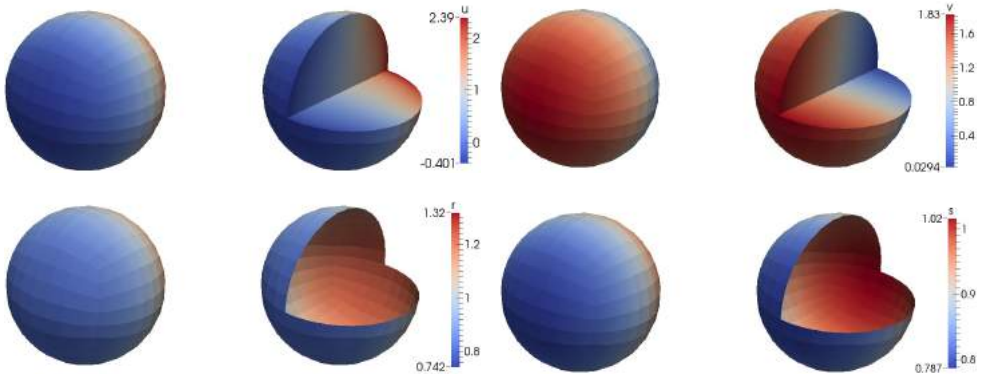


Figure 3.16: Numerical solution to the coupled system of BSRDEs (3.2.1) - (3.2.4) using the parameter values shown in Table 3.3 with $\varepsilon = 10$

3.6 Conclusion, discussion and future research challenges

We have presented a coupled system of bulk-surface reaction-diffusion equations whereby the bulk and surface reaction-diffusion systems are coupled through Robin-type boundary conditions. Nonlinear reaction-kinetics are considered in the bulk and on the surface and for illustrative purposes, the activator-depleted model was selected since it has a unique positive steady state. By using linear stability theory close to the bifurcation point, we state and prove a generalisation of the necessary conditions for Turing diffusion-driven instability for the coupled system of BSRDEs. Our most revealing result is that the bulk reaction-diffusion system has the capability of inducing patterning (under appropriate model and compatibility parameter values) for the surface reaction-diffusion model. On the other hand, the surface reaction-diffusion is not capable of inducing patterning everywhere in the bulk; patterns can only be induced in regions close to the surface membrane. For skin pattern formation, this example is consistent with the observation that patterns will form on the surface as well as within the epidermis layer close to the surface. We do not expect patterning to form everywhere in the body of the animals.

Our studies reveal the following observations and research questions still to be addressed:

- Our numerical experiments reveal that the Robin-type boundary conditions seem to introduce a boundary layer coupling the bulk and surface dynamics. However, these boundary conditions do not appear explicitly in the conditions for diffusion-driven instability and this makes it difficult to theoretically analyse their role and implications on pattern formation. Further studies are required to understand the role of these boundary conditions as well as the size of the boundary layer. The results in Section 3.4.1 shows the behaviour of the bulk solution near the boundary. Further investigation is needed to understand the behaviour fully.

- The compatibility condition (3.2.14) implies that the uniform steady state in the bulk is identical to the uniform state on the surface. We are currently studying the implications of relaxing the compatibility condition.

- Finally, in this chapter, we have not carried out detailed parameter search and estimation to deduce the necessary and sufficient conditions for pattern generation as well isolating excitable wavenumbers in the bulk and on the surface. Such studies

might reveal more interesting properties of the coupled bulk-surface model and this forms part of our current studies.

We have presented a framework that couples bulk dynamics ($3D$) to surface dynamics ($2D$) with the potential of numerous applications in cell motility, developmental biology, tissue engineering and regenerative medicine and biopharmaceutical where reaction-diffusion type models are routinely used (Chechkin et al. 2012, Elliott et al. 2013, Medvdev & Stuchebrukhov 2013, Nisbet et al. 2009, Novak et al. 2007, Rätz & Röger 2012, 2014, Venkataraman et al. 2011).

We have restricted our studies to stationary volumes. In most cases, biological surfaces are known to evolve continuously with time. This introduces extra complexities to the modelling, analysis and simulation of coupled systems of bulk-surface reaction-diffusion equations. In order to consider evolving bulk-surface partial differential equations, evolution laws (geometric) should be formulated describing how the bulk and surface evolve. Here, it is important to consider specific experimental settings that allow for detailed knowledge of properties (biomechanical) and processes (biochemical) involved in the bulk-surface evolution. Such a framework will allow us to study 3D cell migration in the area of cell motility (Elliott et al. 2013, George et al. 2013, Madzvamuse & George 2013, Neilson, Mackenzie, Webb et al. 2011). In future studies, we propose to develop a 3D integrative model that couples bulk and surface dynamics during growth development or movement.

What implication do these results thus have for a model of atherosclerosis? We have seen that when there are excited modes in the bulk system, the bulk is able to generate patterns on the surface, but the reverse is not generally true as the surface system cannot generate patterns everywhere in the bulk - perhaps then we can make a preliminary hypothesis: inflammation in the lipid core may exacerbate inflammation on its cap, but inflammation on the cap does not exacerbate inflammation in the lipid core to the same degree. With the presence of inflammation found in both cap and core of vulnerable plaques (Silvestre-Roig et al. 2014), it may be more likely that the inflammation began in the core. Another implication of this hypothesis is that even if we stop the inflammation at the cap, it will not stop the inflammation in the core, and so drug treatments aimed at abating inflammation should aim to work at the lipid core. Further work can aim to answer questions such as these.

Chapter 4

A Shape Identification Problem

4.1 Introduction

As we have seen in Section 1.6 one of the first tools a clinician uses in assaying a patient with atherosclerosis is an ultrasound scan. This is often done to examine the carotid bifurcation since this is where atherosclerosis is most prevalent and, due to its location it is amenable to ultrasound scanning. It is known however that the performance and the interpretation of ultrasound scans are operator-dependent in nature (Mikkonen et al. 1996, Alexandrov et al. 1997) and the use of ultrasound alone can lead to disagreement in the treatment of atherosclerosis (Meadb et al. 2000). Therefore, there exists a body of research dedicated to automating the interpretation of ultrasound scans (Liguori et al. 2001, Faita et al. 2008, Molinari et al. 2012, Sabaa et al. 2013, Cheng et al. 2013, Sundholma et al. 2014, Menchón-Lara & Sancho-Gómez 2015, Akkus et al. 2015, Molinari et al. 2010). The goal is the automatic segmentation of the pictures into the areas of lumen, intima, media, adventitia and plaque which automatically provide important measurements such as the intima-media thickness (see Section 1.6) in the hope of eliminating inter-operator variability. The methods employed use the pixel intensity profile of the image with imaging techniques such as edge detection, statistics, optimisation and machine learning (Molinari et al. 2010). Many results published have been validated using real data with comparisons made between the software's and expert radiologist's measurements of the intima-media thickness. These differences range from about 10μm to about 100μm (Menchón-Lara & Sancho-Gómez 2015).

In using the pixel intensities, each method has to overcome the inherent noise in ultrasound data, known as speckle (Burckhardt 1978, Damerjian et al. 2014). This gives

the image a granular appearance and mars the overall quality of the ultrasound picture (Burckhardt 1978). Yet, the quality of ultrasound imaging continues to improve with new despeckling techniques (El-said & Azar 2012). Therefore, let us approach the problem from a different angle: rather than use the pixel intensities, let us use the velocity profiles - which are data obtainable by dupluex ultrasound scanning - to identify the shape of the artery wall. The problem is thus an inverse problem: given the inflow and outflow velocity profiles, can we identify the shape of the artery walls? This has been addressed to some extent in the literature (Toivanen et al. 2008, Kasumba & Kunisch 2012, Michel et al. 2014). However, in the context of atherosclerosis, this approach is not documented. Inverse problems have the difficulty of not having, in general, a unique solution. However, in this chapter, we shall see that the methods we use seem to lead to reasonable solutions.

In this chapter, we start this approach by considering a free boundary problem whose solution we obtain via PDE-constrained optimisation. We shall use a gradient descent method to obtain numerical solutions and test our approach with a constructed solution.

4.2 Statement of the Problem

4.2.1 Basic notation

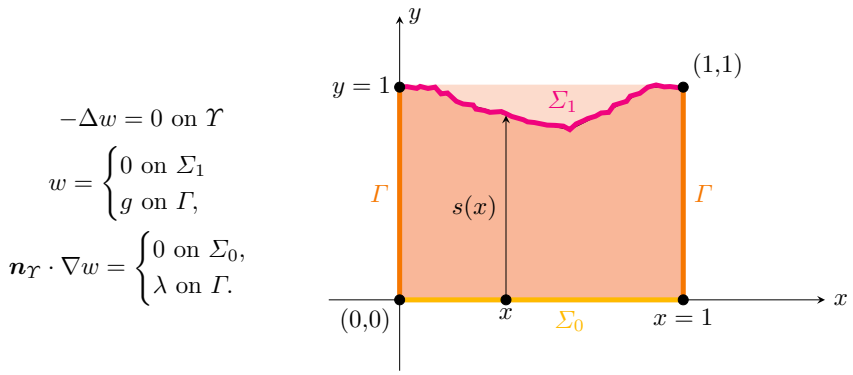
Given $d, k \in \mathbb{N}$ and a (Lebesgue or Hausdorff) measurable set $D \subseteq \mathbb{R}^d$ we and two functions $f, g \in L_2(D)^k$ we denote by

$$\langle \mathbf{f}, \mathbf{g} \rangle := \int_D \mathbf{f}(\mathbf{x}) \cdot \mathbf{g}(\mathbf{x}) \mu(\mathrm{d}\,\mathbf{x}) \tag{4.2.1}$$

where μ is either the volume (Lebesgue) or the surface ($d-1$ dimensional Hausdorff) measure depending on the nature of D and $\cdot$ denotes the inner product in $\mathbb{R}^k$.

4.2.2 The axially symmetric problem

Let $\Sigma_0 := [0,1] \times \{0\} \subset \mathbb{R}^2$, $\Gamma := \cup_{x=0,1}\{x\} \times (0,1) \subset \mathbb{R}^2$ and $\Sigma_1 \subset \mathbb{R}^2$ be an unknown boundary with endpoints $(0,1)$ and $(1,1)$. Let $\Upsilon \subset \mathbb{R}^2$ be the area enclosed by the boundaries, such that $\partial\Upsilon = \Gamma \cup \Sigma_0 \cup \Sigma_1$ disjointly. Consider the following *free boundary*



problem (see Fig. 4.2.2): find a function w and a curve Σ_1 satisfying

$$\begin{aligned} -\Delta w &= 0 \text{ on } \Upsilon \\ w &= \begin{cases} 0 \text{ on } \Sigma_1 \\ g \text{ on } \Gamma, \end{cases} \\ \mathbf{n}_\Upsilon \cdot \nabla w &= \begin{cases} 0 \text{ on } \Sigma_0, \\ \lambda \text{ on } \Gamma. \end{cases} \end{aligned} \tag{4.2.2}$$

Thus Υ is an unknown domain depending on the unknown boundary Σ_1. The data g and λ are given functions, which we will discuss in detail further.

If Σ_1 were given rather than unknown, Problem (4.2.2) would be ill-posed. Indeed, there are too many boundary conditions (on Γ) which makes it overdetermined. This problem will be solvable only if the side Σ_1 is left unknown as a free boundary. Thus, we are dealing with an inverse problem where we need to identify the shape of the domain Υ for the overdetermined PDE (4.2.2) to admit (at least) one solution.

One way to approach the solvability of this inverse problem is that of *shape-identification*. To formulate a shape-identification approach we will make an extra assumption on Σ_1.

Assumption 1. *(graph boundary) We assume hereafter that the curve Σ_1 is given as a graph of a function $s : [0,1] \to \mathbb{R}$,*

$$\Sigma_1 := \{(x,y) : 0 < x < 1 \text{ and } y = s(x)\} \tag{4.2.3}$$

satisfying the following conditions

$$s \in W^{1,\infty}(0,1),\ 0 < s_0 \le s \le 1 \text{ for some } s_0 \in (0,1),\ \text{and } s(0) = 1 = s(1). \tag{4.2.4}$$

4.2.3 An overview of the approach

Our main objective is to design a fully practical numerical method to approximate (w, Σ_1) statisfying (4.2.2). We approach this by considering the case where Σ_1 is the graph of a function s (see Assumption 1). By taking the change of coordinates

$$u(t,z) = w(x,y) \text{ with } x = t \text{ and } y = s(t)z, \tag{4.2.5}$$

we turn the free boundary Problem (4.2.2) into a free coeffcients PDE (see Section 4.2.4), which can be written in the strong form as follows: find a pair of functions (s,u), such that

$$\begin{gathered} u : (0,1)^2 \to \mathbb{R} \text{ and } s : (0,1) \to \mathbb{R} \\ -\nabla \cdot \begin{bmatrix} s(t) & -s'(t)z \\ -s'(t)z & \dfrac{1+s'(t)^2z^2}{s(t)} \end{bmatrix} \begin{bmatrix} \partial_z u(t,z) \\ \partial_t u(t,z) \end{bmatrix} = 0 \text{ for } (z,t) \in \Omega := (0,1)^2, \end{gathered} \tag{4.2.6}$$

with the following boundary conditions

$$\begin{gathered} s(0) = s(1) = 1 \\ u = \begin{cases} 0 & \text{on } (0,1) \times \{1\} =: \widetilde{\Sigma}_1 \\ g & \text{on } \{0,1\} \times (0,1) = \Gamma \end{cases} \\ \mathbf{n}_\Omega \cdot \mathbf{A}_s \nabla u = \begin{cases} 0 & \text{on } (0,1) \times \{0\} = \Sigma_0 \\ \lambda & \text{on } \{0,1\} \times (0,1) = \Gamma \end{cases} \end{gathered} \tag{4.2.7}$$

4.2.4 Turning a free boundary problem into a free coefficients PDE

Assumption 1 allows us to transform the free boundary Problem (4.2.2) into a problem with fixed boundary and unknown coefficients instead. This is achieved by simply replacing the geometric unknown Σ_1 by its analytic counterpart s. With respect to the numerical methods developed further, this transformation can be performed in two alternative (but clearly equivalent) ways:

A. Transform coordinates first and then apply a numerical method, possibly a Galerkin one based on the weak formulation.

B. Write a weak formulation first and then transform the coordinates.

We choose alternative B as it allows for Galerkin methods to be applied easily, whereas A does not. So, let us start by writing problem (4.2.2) in weak form. Test the equation with a smooth function $\psi \in C^\infty(\Upsilon)$ which vanishes on Σ_1, and use the boundary conditions, to get

$$\begin{aligned} 0 &= -\int_\Upsilon \Delta w(x,y)\psi(x,y)\,\mathrm{d}(x,y) \\ &= \int_\Upsilon \nabla w(x,y)\cdot\nabla\psi(x,y)\,\mathrm{d}(x,y) + \int_{\partial\Upsilon} \mathbf{n}_\Upsilon\cdot\nabla w(x,y)\psi(x,y)\,\mathrm{dS}\,(x,y) \\ &= \langle\nabla w,\nabla\psi\rangle_\Upsilon + \langle \mathbf{n}\cdot\nabla w,\psi\rangle_\Gamma \\ &= \langle\nabla w,\nabla\psi\rangle_\Upsilon + \langle\lambda,\psi\rangle_\Gamma. \end{aligned} \tag{4.2.8}$$

It follows that, for a given Σ_1, a smooth function w satisfies (4.2.2) if and only if it satisfies

$$w|_\Gamma = g \tag{4.2.9}$$

and

$$\langle\nabla w,\nabla\psi\rangle_\Upsilon + \langle\lambda,\psi\rangle_\Gamma = 0 \quad \forall\psi\in C^\infty(\Upsilon) : \psi|_{\Sigma_1} = 0. \tag{4.2.10}$$

To introduce the coordinate change we use the function s of Assumption 1 and the following coordinate transformation (change of variables)

$$\begin{bmatrix} t \\ z \end{bmatrix} = \mathbf{T}(x,y) := \begin{bmatrix} x \\ \dfrac{y}{s(x)} \end{bmatrix} \tag{4.2.11}$$

which transforms the domain Υ into a *new domain*

$$\mathbf{T}(\Upsilon) = (0,1)^2 =: \Omega \text{ and } \mathbf{T}(\partial\Upsilon) = \partial\Omega. \tag{4.2.12}$$

We consider thus the substitution

$$u(\mathbf{T}(x,y)) = w(x,y) \text{ and } \varphi(\mathbf{T}(x,y)) = \psi(x,y) \tag{4.2.13}$$

in the bilinear forms in (4.2.10). To operate with $\mathbf{T}$ we need its Jacobian

$$\mathrm{D}\,\mathbf{T}(x,y) = \begin{bmatrix} 1 & 0 \\ -\dfrac{s'(x)y}{s(x)^2} & \dfrac{1}{s(x)} \end{bmatrix} \text{ and } \det\mathrm{D}\,\mathbf{T}(x,y) = \frac{1}{s(x)}. \tag{4.2.14}$$

Hence we have $\det\mathrm{D}[\mathbf{T^{-1}}(t,z)] = s(t)$. We will also need the matrix

$$\mathbf{J}(t,z) := \mathrm{D}\,\mathbf{T}(\mathbf{T^{-1}}(t,z)) = \begin{bmatrix} 1 & 0 \\ -\dfrac{s'(t)z}{s(t)} & \dfrac{1}{s(t)} \end{bmatrix} \tag{4.2.15}$$

and its *symmetric square*

$$\mathbf{J}(t,z)\mathbf{J}(t,z)^{\intercal} = \begin{bmatrix} 1 & -\dfrac{s'(t)z}{s(t)} \\ -\dfrac{s'(t)z}{s(t)} & \dfrac{1+s'(t)^2z^2}{s(t)^2} \end{bmatrix}. \tag{4.2.16}$$

Using the chain rule first and then change of variables

$$\begin{aligned} \langle \nabla w, \nabla \psi \rangle_{\Upsilon} &= \int_{\Upsilon} \nabla \left[u\left(\mathbf{T}(x,y)\right) \right] \cdot \nabla \left[\varphi\left(\mathbf{T}(x,y)\right) \right] \mathrm{d}(x,y) \\ &= \int_{\Upsilon} \nabla u\left(\mathbf{T}(x,y)\right) \mathrm{D}\,\mathbf{T}(x,y)\, \mathrm{D}\,\mathbf{T}(x,y)^{\intercal} \nabla \varphi(\mathbf{T}(x,y)) \,\mathrm{d}(x,y) \\ &= \int_{\Omega} \nabla u(t,z)\mathbf{J}(t,z)\mathbf{J}(t,z)^{\intercal} \nabla \varphi(t,z)\,\mathrm{d}(t,z) \\ &= \langle \mathbf{A_s} \nabla u, \nabla \varphi \rangle_{\Omega} \\ &=: a_s[u,\varphi], \end{aligned} \tag{4.2.17}$$

where

$$\mathbf{A_s}(t,z) := \begin{bmatrix} s(t) & -s'(t)z \\ -s'(t)z & \dfrac{1+s'(t)^2z^2}{s(t)} \end{bmatrix}. \tag{4.2.18}$$

To simplify notation we shall denote this relation as

$$\begin{gathered} \mathbf{A_s}(t,z) = \hat{\mathbf{A}}(s(t), s'(t), z), \text{ where } \hat{\mathbf{A}} : \mathbb{R}_+ \times \mathbb{R} \times \mathbb{R} \to \mathbb{R}^{2\times 2} \text{ such that} \\ \hat{\mathbf{A}}(x,y,z) := \begin{bmatrix} x & -yz \\ -yz & \dfrac{1+y^2z^2}{x} \end{bmatrix}. \end{gathered} \tag{4.2.19}$$

The boundary integral, thanks to Assumption 1, which implies that $\mathbf{T}|_{\Gamma} = \mathbf{I}$, is transformed trivially:

$$\begin{aligned} \langle \lambda, \psi \rangle_{\Gamma} &= \int_{\Gamma} \lambda(x,y)\psi(x,y)\,\mathrm{dS}\,(x,y) = \int_{\Gamma} \lambda(t,z)\varphi(t,z)\,\mathrm{dS}\,(t,z) \\ &= \langle \lambda, \varphi \rangle_{\Gamma}. \end{aligned} \tag{4.2.20}$$

Similarly the overdetermining boundary condition $w|_{\Gamma} = g$ transforms into

$$u|_{\Gamma} = g. \tag{4.2.21}$$

4.2.5 A fixed boundary formulation of Problem (4.2.2)

With these transformations at hand we can reformulate a weak form equivalent problem to Problem (4.2.2) as follows:

$$\begin{aligned}
&\text{Find } s \in \mathscr{W}, \text{ and } u \in \mathscr{U} \text{ such that} \\
&s(0) = s(1) = 1 \text{ and} && (4.2.22)\\
&a_s[u,\varphi] = \langle \lambda, \varphi \rangle_\Gamma \quad \forall \varphi \in \mathscr{U} && (4.2.23)\\
&u|_\Gamma = g && (4.2.24)
\end{aligned}$$

where $a_s[\cdot,\cdot]$ is defined in (4.2.17) and the functional spaces are given by

$$\begin{aligned}
\mathscr{W} &:= \{\chi \in \mathrm{L}^2(0,1) : \chi' \in \mathrm{L}(0,1)\} \\
\mathscr{U} &:= \{\varphi \in \mathrm{H}^1(\Omega) : \varphi|_{\{1\}\times(0,1)} = 0\}.
\end{aligned} \tag{4.2.25}$$

4.3 Constrained optimization and regularization

4.3.1 Definitions

We introduce the solution operator (also known as *Green's operator*)

$$\hat{u}[s] := \text{solution of (4.2.23) for a given } s \in \mathscr{W}, \tag{4.3.1}$$

the boundary energy form

$$\mathscr{I}[s,u] := \int_\Gamma (u-g)^2 + \varepsilon_s \int_{\Sigma_0} s'^2 + \varepsilon_u \int_\Omega u^2, \tag{4.3.2}$$

where $\varepsilon_s > 0$ is a *regularization parameter* and $\varepsilon_u > 0$ is a *stability parameter*, and the related solution-composed energy functional

$$\mathscr{J}[s] := \mathscr{I}[s, \hat{u}[s]]. \tag{4.3.3}$$

We define heareafter the *Lagrangian energy* associated with the problem (4.2.22)–(4.2.23):

$$\mathscr{L}[s,u;p] = \mathscr{I}[s,u] + a_s[u,p] - \langle \lambda, p \rangle_\Gamma, \tag{4.3.4}$$

for $s \in \mathscr{W}$, $u \in \mathscr{U}$ and $p \in \mathscr{U}$. .

4.3.2 Optimality system

Suppose $(s,u,p) \in \mathscr{W} \times \mathscr{U} \times \mathscr{U}$ satisfy the Euler-Lagrange equations associated with $\mathscr{L}$, i.e., the so-called *optimality system*

$$\partial_s \mathscr{L}[s,u;p] = 0, \quad \partial_u \mathscr{L}[s,u;p] = 0 \text{ and } \partial_p \mathscr{L}[s,u;p] = 0, \tag{4.3.5}$$

where the partial derivatives are understood in the usual (Fréchet) sense, as elements of the dual space to the differentiation argument. The system can be made concrete by interpreting it with directional (Gâteaux) derivatives, which we proceed in doing now.

In particular, if (s,u,p) satisfies (4.3.5) then (s,u) is a local solution of (4.2.23) in the sense that u satisfies $a_s[u,\varphi] - \langle \lambda, \varphi \rangle_\Gamma = 0$ for all $\varphi \in \mathscr{U}$ and $\mathscr{I}$ reaches a local minimum in (s,u).

To see this we give a more explicit representation of the optimality system (4.3.5). For convenience we start with the last equation, which can be written as an elliptic equation for u (4.2.23) by noting that

$$a_s[u,\varphi] - \langle \lambda, \varphi \rangle_\Gamma = \langle \partial_p \mathscr{L}[s,u;p] \,|\, \varphi \rangle_{\mathscr{U}' \times \mathscr{U}} = 0 \quad \forall \varphi \in \mathscr{U}. \tag{4.3.6}$$

Next the second equation in (4.3.5) takes the form

$$\begin{aligned} 0 &= \langle \partial_u \mathscr{L}[s,u;p] \,|\, \psi \rangle_{\mathscr{U}' \times \mathscr{U}} \\ &= \langle \partial_u \mathscr{I}[s,u] \,|\, \psi \rangle_{\mathscr{U}' \times \mathscr{U}} + \langle \partial_u a_s[u,p] \,|\, \psi \rangle_{\mathscr{U}' \times \mathscr{U}} \\ &= \langle 2(u-g), \psi \rangle_\Gamma + a_s[\psi,p] + 2\varepsilon_u \langle u, \psi \rangle_\Omega \quad \forall \psi \in \mathscr{U}, \end{aligned} \tag{4.3.7}$$

which we may rewrite in the following *adjoint equation* for p form

$$a_s^*[p,\psi] = -2\langle u-g, \psi \rangle_\Gamma - 2\varepsilon_u \langle u, \psi \rangle_\Omega \quad \forall \psi \in \mathscr{U}, \tag{4.3.8}$$

where

$$a_s^*[\varphi,\psi] := a_s[\psi,\varphi] \text{ for } \psi \in \mathscr{U},\ \varphi \in \mathscr{U}. \tag{4.3.9}$$

Finally, the first equation in (4.3.5) is made explicit as follows:

$$0 = \langle \partial_s \mathscr{L}[s,u;p] \,|\, \chi \rangle = 2\varepsilon_s \langle s', \chi' \rangle + \langle \partial_s a_s[u,p] \,|\, \chi \rangle. \tag{4.3.10}$$

We have

$$\begin{aligned} &\langle \partial_s a_s[u,p] \,|\, \chi \rangle = \langle \partial_s \langle \mathbf{A_s} \nabla u \cdot \nabla p \rangle_\Omega \,|\, \chi \rangle \\ &= \langle \partial_s \int_\Omega \hat{\mathbf{A}}(s(t), s'(t), z) \nabla u(t,z) \cdot \nabla p(t,z) \,\mathrm{d}\, t \,\mathrm{d}\, z \,|\, \chi \rangle \\ &= \int_\Omega \Big(\chi(t) \partial_1 \hat{\mathbf{A}}(s(t), s'(t), z) + \chi'(t) \partial_2 \hat{\mathbf{A}}(s(t), s'(t), z) \Big) \nabla u(t,z) \cdot \nabla p(t,z) \,\mathrm{d}\, t \,\mathrm{d}\, z \end{aligned} \tag{4.3.11}$$

and using (4.2.19) we have

$$\partial_1 \hat{\mathbf{A}}(x,y,z) := \partial_x \hat{\mathbf{A}}(x,y,z) = \begin{bmatrix} 1 & 0 \\ 0 & -\dfrac{1+y^2z^2}{x^2} \end{bmatrix} =: \hat{\mathbf{B}}^0(x,y,z), \tag{4.3.12}$$

$$\partial_2 \hat{\mathbf{A}}(x,y,z) := \partial_y \hat{\mathbf{A}}(x,y,z) = \begin{bmatrix} 0 & -z \\ -z & \dfrac{2yz^2}{x} \end{bmatrix} =: \hat{\mathbf{B}}^1(x,y,z) \tag{4.3.13}$$

whence equation (4.3.10) can be written explicitly as a *nonlinear second order elliptic equation* for $s \in \mathscr{W}$ in the following weak form

$$\langle 2\varepsilon_s s' + \nabla p \cdot \mathbf{B}_s^1 \nabla u, \chi' \rangle + \langle \nabla p \cdot \mathbf{B}_s^0 \nabla u \rangle \chi = 0 \quad \forall \chi \in \mathscr{W}_0 \tag{4.3.14}$$

where

$$\mathbf{B}_s^i(t,z) = \hat{\mathbf{B}}^i(s(t), s'(t), z), \text{ for } i = 0, 1. \tag{4.3.15}$$

In the rest of the chapter we study numerical methods to solve the optimality system (4.3.5).

4.4 A descent method

Our first line of attack to the numerical approximation of the problem is inspired by its variational nature. Namely, we perform an operator splitting and treat the nonlinearity via a descent method.

4.4.1 An iterative descent algorithm

From this observation we present an iterative algorithm for the problem (4.2.23). Let $k \in \mathbb{N}$. Given $s^{k-1} \in \mathscr{W}$ and $p^{k-1} \in \mathscr{U}$, and $u^{k-1} \in \mathscr{U}$ we proceed to find (s^k, u^k, p^k) as follows:

sub-step 1. Find $u^k \in \mathscr{U}$ such that

$$\partial_p \mathscr{L}[s^{k-1}, u^k; p^{k-1}] = 0. \tag{4.4.1}$$

Written weakly, this problem takes the form: find $u^k \in \mathscr{U}$ satistying

$$a_{s^{k-1}}[u^k, \varphi] - \langle \lambda, \varphi \rangle = \langle \partial_p \mathscr{L}[s^{k-1}, u^k; p^{k-1}] \,|\, \varphi \rangle_{\mathscr{U}' \times \mathscr{U}} = 0 \quad \forall \varphi \in \mathscr{U}. \tag{4.4.2}$$

Note that the left-hand-side of (4.4.1) and (4.4.2) does not depend on p^{k-1} (for the linearity in p of the Lagrangian $\mathscr{L}[s,u;p]$) .

sub-step 2. Find $p^k \in \mathscr{U}$ such that

$$\partial_u \mathscr{L}[s^{k-1}, u^k; p^k] = 0 \in \mathscr{U}', \tag{4.4.3}$$

which can be written in the weak form as

$$\begin{aligned} 0 &= \langle \partial_u \mathscr{L}[s^{k-1}, u^k; p^k] \,|\, \psi \rangle_{\mathscr{U}' \times \mathscr{U}} \\ &= \langle \partial_u \mathscr{I}[s^{k-1}, u^k] \,|\, \psi \rangle_{\mathscr{U}' \times \mathscr{U}} + \langle \partial_u a_{s^{k-1}}[u^k, p^k] \,|\, \psi \rangle_{\mathscr{U}' \times \mathscr{U}} \\ &= \langle 2(u^k - g), \psi \rangle_\Gamma + a_{s^{k-1}}[\psi, p^k] + 2\varepsilon_u \langle u^k, \psi \rangle_\Omega \quad \forall \psi \in \mathscr{U}. \end{aligned} \tag{4.4.4}$$

Recalling (4.3.9), the last term appearing in (4.4.4) can be rewritten and we obtain the following *adjoint problem*: find $p^k \in \mathscr{U}$ such that

$$a^*_{s^{k-1}}[p^k, \psi] = -2\langle u^k - g, \psi \rangle_\Gamma - 2\varepsilon_u \langle u^k, \psi \rangle_\Omega \quad \forall \psi \in \mathscr{U}. \tag{4.4.5}$$

sub-step 3. Define the *descent direction* δ^k as a function in $\mathscr{W}_0$ satisfying

$$\langle (\delta^k)', \chi' \rangle = -\langle \partial_s \mathscr{L}[s^{k-1}, u^k; p^k] \,|\, \chi \rangle_{\mathscr{W}_0' \times \mathscr{W}_0} \quad \forall \chi \in \mathscr{W}_0 \tag{4.4.6}$$

$$= -\langle 2\varepsilon_s (s^{k-1})' + \nabla p^k \cdot \mathbf{B}^1_{s^{k-1}} \nabla u^k, \chi' \rangle - \langle \nabla p \cdot \mathbf{B}^0_{s^{k-1}} \nabla u \rangle \chi. \tag{4.4.7}$$

Then, we find

$$\rho^k := \inf \operatorname*{argmin}_{\rho > 0} \mathscr{J}(s^{k-1} + \rho \delta^k), \tag{4.4.8}$$

i.e. ρ^k is the first ρ that makes $s^{k-1} + \rho\delta^k$ a local minimiser of $\mathscr{J}$. And finally, we define the new solution $s^k = s^{k-1} + \rho^k \delta^k$.

4.5 Computational Implementation

During the simulations, to find the step size ρ^k we find the minimum of the solution composed energy functional, $\mathscr{J}$

$$\rho^k := \inf \operatorname*{argmin}_{\rho > 0} \mathscr{J}[s^k + \rho \delta^k]. \tag{4.5.1}$$

This is done using the secant method, i.e. given ρ^k_0 and ρ^k_1, we solve iteratively

$$\rho^k_l = \rho^k_{l-1} - \mathscr{J}'[s^k + \rho^k_{l-1}\delta^k] \frac{\rho^k_{l-1} - \rho^k_{l-2}}{\mathscr{J}'[s^k + \rho^k_{l-1}\,\delta^k] - \mathscr{J}'[s^k + \rho^k_{l-2}\,\delta^k]}, \tag{4.5.2}$$

and we use a central difference approximation to the derivative,

$$\mathscr{J}'[s^k + \rho^k_{l-1}\delta^k] \approx \frac{\mathscr{J}[s^k + (\rho^k_{l-1} + h)\,\delta^k] - \mathscr{J}[s^k + (\rho^k_{l-1} - h)\,\delta^k]}{2h}, \tag{4.5.3}$$

for some appropriate small h. This involves solving for two more solutions $\hat{u}_h[s^k + (\rho^k_{l-1} + h)\,\delta^k]$ and $\hat{u}_h[s^k + (\rho^k_{l-1} - h)\,\delta^k]$ at each iteration. This is a drawback to calculating the derivative this way and is a bottleneck in the code.

4.6 Numerical results

We test the algorithm in the following way: choose a function $s_{ex}(t)$ and $\lambda(t,z)$. Solve for $\bar{u}(t,z)$

$$\begin{cases} -\nabla\cdot\big(\mathbf{A}_{s_{ex}}\nabla\bar{u}(t,z)\big)=0 & \text{in } \Omega \\ \bar{u}(t,z)=0 & \text{on } \Sigma_1 \\ \mathbf{n}\cdot\nabla\bar{u}(t,z)=0 & \text{on } \Sigma_0 \\ \mathbf{n}\cdot\nabla\bar{u}(t,z)=\lambda(t,z) & \text{on } \Gamma\,. \end{cases} \tag{4.6.1}$$

We run the minimisation algorithm choosing an initial guess $s_0(t)$ and using as boundary conditions the same $\lambda(t,z)$ and the above solution $\bar{u}(x,y)$ restricted to Γ for g. The hope is that the $s_k(t)$ converge to the chosen function $s_{ex}(t)$.

To be concrete, for the exact solution (see Fig. 4.6) we choose one of three expressions: a smooth symmetrical curve

$$s_{ex}(t)=1-8t^2(1-t)^2, \tag{4.6.2}$$

a smooth asymmetrical curve

$$s_{ex}(t)=1-5t^3(1-t), \tag{4.6.3}$$

or a "v-shaped" domain

$$s_{ex}(t)=\frac{1+|2t-1|}{2}. \tag{4.6.4}$$

For the boundary data we choose

$$\lambda(t,z)=z(z-1), \tag{4.6.5}$$

and for the initial guess $s_0(t)$ (see Fig. 4.6) of the algorithm we choose

$$s_0(t)=1+\mu_1\big(s_{ex}(t)-1\big)+\mu_2 t(1-t)\sin(8\pi t). \tag{4.6.6}$$

We note the special cases

- $\mu_1=0$, $\mu_2=0$: we start from the straight line $s_0(t)=1$
- $\mu_1=1$, $\mu_2=0$: we start from the exact solution $s_0(t)=s_{ex}(t)$
- $\mu_2>0$: we add perturbations to the initial guess.

In the following we take combinations of the following parameters

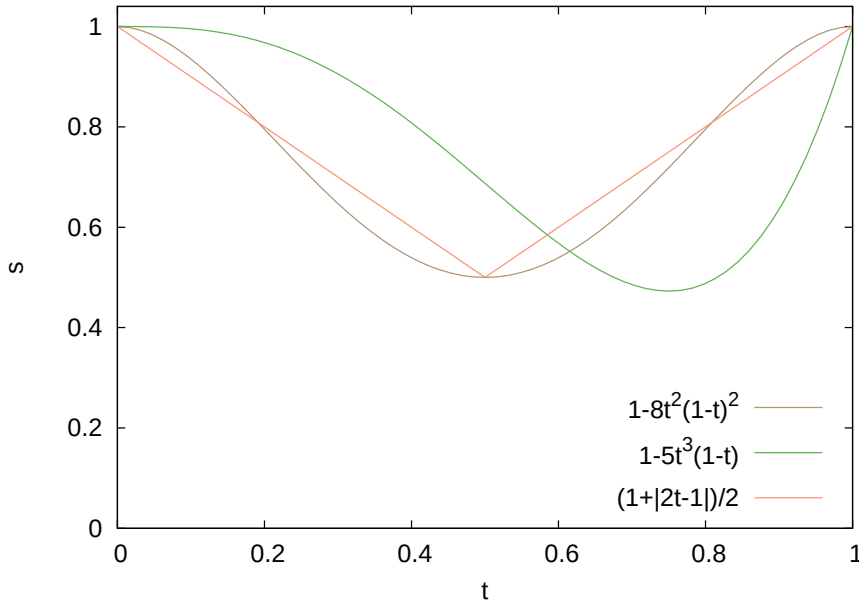


Figure 4.1: The test-cases $s_{ex}(t)$ used for testing the algorithm.

- $\mu_1 = 0, 0.5, 1$
- $\mu_2 = 0, 0.2$
- $\varepsilon_s = 10^{-2}, 10^{-4}, 10^{-6}$
- $\varepsilon_u = 0, 10^{-2}, 10^{-4}$.

We note the number of iterations k needed for the algorithm to stop, and the L_2-norm of the final solution with the exact solution $||s_k(t) - s_{ex}(t)||$. The stopping criterion was chosen to be

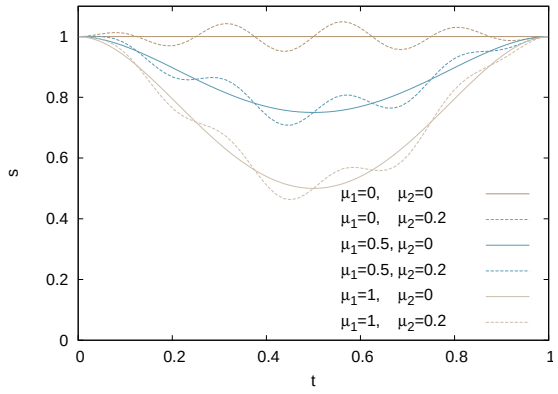
$$||s_k - s_{k-1}|| < 10^{-5}, \tag{4.6.7}$$

with the maximum number of iterations set to 1000. A 64×64 grid, totalling 4096 nodes, was used.

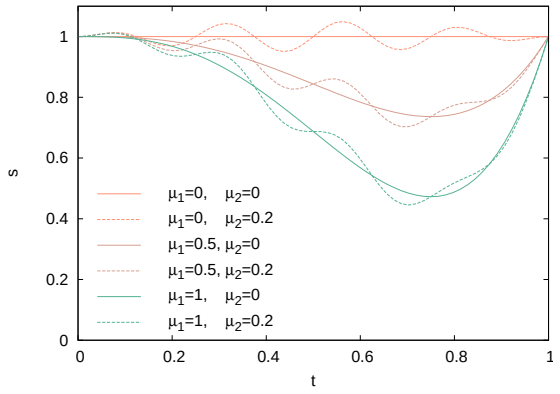
The simulations were carried out on an Intel Core i7-3632QM Processor with 2.20 GHz frequency. The assembly of the matrices in the algorithm was parallelised using 8 threads. The CPU time of operation was recorded and is shown in the results.

We present the results in Tables 4.1-4.9 and set aside the figures in Appendix B. The results are promising and show that with certain values of the parameters, the method converged satisfactorily to the constructed test-case solution. We note the following:

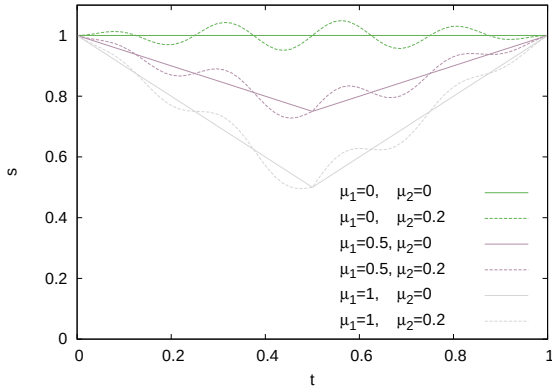
- the value of ε_u which seemed to work best was 10^{-4}. At $\varepsilon_u = 10^{-2}$ the value of



(a) $s_{ex}(t) = 1 - 8t^2(1-t)^2$



(b) $s_{ex}(t) = 1 - 5t^3(1-t)$



(c) $s_{ex}(t) = \frac{1+|2t-1|}{2}$

Figure 4.2: Initial guesses $s_0(t)$ used to start the algorithm (see equation (4.6.6)).

$||s_k - s_{ex}||$ is relatively higher and indeed, from the graphs one can see that the corresponding curves do not always lie as close to the desired result. When $\varepsilon_u = 0$ no noticeable gain in accuracy is seen in the values of $||s_k - s_{ex}||$ compared to 10^{-4}. It is also seen, in this case, that the time taken for the simulation was higher in comparison.

- in the absence of perturbations in the initial guess ($\mu_2 = 0$) the choice $\varepsilon_s = 10^{-6}$ worked best, giving the lowest values of $||s_k - s_{ex}||$ and requiring the least number of iterations and CPU time. However, when we add perturbations ($\mu_2 = 0.2$) the best choice is $\varepsilon_s = 10^{-4}$. In this case, taking $\varepsilon_s = 10^{-6}$ does not sufficiently rid the final solution $s_k(t)$ of its wave-like nature.

- these observations hold with each test-case solution constructed. In particular, the method converged reasonably close in the cases where the test-case solution was either asymmetrical or even non-smooth continuous.

μ_1	μ_2	ε_s	ε_u	k	CPU time ($\times 10^2$s)	$\mathcal{I}$	$\|\|s_k - s_{k-1}\|\|$	$\|\|s - s_{ex}\|\|$
0	0	10^{-2}	10^{-2}	10	10.70	5.34×10^{-03}	0.00	2.54×10^{-01}
			10^{-4}	11	9.803	5.34×10^{-03}	3.09×10^{-06}	2.54×10^{-01}
			0	17	12.35	5.34×10^{-03}	2.53×10^{-06}	2.54×10^{-01}
		10^{-4}	10^{-2}	34	35.05	1.11×10^{-04}	9.83×10^{-06}	3.95×10^{-02}
			10^{-4}	93	67.92	1.11×10^{-04}	4.30×10^{-06}	3.28×10^{-02}
			0	142	108.0	1.11×10^{-04}	7.93×10^{-06}	3.31×10^{-02}
		10^{-6}	10^{-2}	53	353.6	2.24×10^{-06}	6.51×10^{-08}	1.86×10^{-02}
			10^{-4}	557	280.4	1.22×10^{-06}	3.41×10^{-06}	4.88×10^{-03}
			0	1000	426.4	1.22×10^{-06}	7.07×10^{-05}	3.30×10^{-03}
	0.2	10^{-2}	10^{-2}	7	8.744	5.34×10^{-03}	2.95×10^{-07}	2.54×10^{-01}
			10^{-4}	15	16.01	5.34×10^{-03}	3.08×10^{-07}	2.54×10^{-01}
			0	16	11.69	5.34×10^{-03}	2.03×10^{-09}	2.54×10^{-01}
		10^{-4}	10^{-2}	79	168.7	1.13×10^{-04}	0.00	3.91×10^{-02}
			10^{-4}	8	15.21	1.49×10^{-04}	5.43×10^{-01}	5.92×10^{-02}
			0	252	319.3	1.11×10^{-04}	6.62×10^{-06}	3.32×10^{-02}
		10^{-6}	10^{-2}	68	128.9	2.25×10^{-06}	7.13×10^{-08}	9.78×10^{-02}
			10^{-4}	168	212.3	1.78×10^{-06}	7.39×10^{-06}	9.18×10^{-02}
			0	175	255.4	1.81×10^{-06}	8.22×10^{-06}	9.29×10^{-02}

Table 4.1: Results on a 64×64 grid (4096 nodes) with $\mu_1 = 0$ with $s_{ex}(t) = 1 - 8t^2(1-t)^2$. Graphs shown in Fig. B.1

μ_1	μ_2	ε_s	ε_u	k	CPU time ($\times 10^2$s)	$\mathcal{I}$	$\|\|s_k - s_{k-1}\|\|$	$\|\|s - s_{ex}\|\|$
0.5	0	10^{-2}	10^{-2}	17	17.37	5.34×10^{-03}	4.37×10^{-11}	2.53×10^{-01}
			10^{-4}	14	10.38	5.34×10^{-03}	3.96×10^{-07}	2.54×10^{-01}
			0	16	13.41	5.34×10^{-03}	2.55×10^{-06}	2.54×10^{-01}
		10^{-4}	10^{-2}	3	4.145	1.11×10^{-04}	6.53×10^{-08}	3.60×10^{-02}
			10^{-4}	78	65.19	1.11×10^{-04}	4.38×10^{-06}	3.29×10^{-02}
			0	140	110.6	1.11×10^{-04}	5.14×10^{-06}	3.31×10^{-02}
		10^{-6}	10^{-2}	23	18.53	1.99×10^{-06}	4.77×10^{-11}	1.63×10^{-02}
			10^{-4}	379	196.0	1.22×10^{-06}	8.10×10^{-06}	3.58×10^{-03}
			0	888	392.9	1.21×10^{-06}	4.87×10^{-06}	2.54×10^{-03}
	0.2	10^{-2}	10^{-2}	6	6.975	5.34×10^{-03}	4.59×10^{-07}	2.53×10^{-01}
			10^{-4}	17	19.73	5.34×10^{-03}	1.87×10^{-07}	2.54×10^{-01}
			0	14	12.24	5.34×10^{-03}	9.25×10^{-09}	2.54×10^{-01}
		10^{-4}	10^{-2}	38	64.94	1.14×10^{-04}	4.25×10^{-06}	4.16×10^{-02}
			10^{-4}	218	291.4	1.11×10^{-04}	4.92×10^{-06}	3.32×10^{-02}
			0	114	189.5	1.11×10^{-04}	5.11×10^{-06}	3.31×10^{-02}
		10^{-6}	10^{-2}	14	29.63	3.10×10^{-06}	1.77×10^{-07}	8.32×10^{-02}
			10^{-4}	161	166.1	1.67×10^{-06}	6.94×10^{-06}	7.55×10^{-02}
			0	277	297.4	1.63×10^{-06}	5.80×10^{-06}	7.43×10^{-02}

Table 4.2: Results on a 64×64 grid (4096 nodes) with $\mu_1 = 0.5$ with $s_{ex}(t) = 1 - 8t^2(1-t)^2$. Graphs shown in Fig. B.1

μ_1	μ_2	ε_s	ε_u	k	CPU time ($\times 10^2$s)	$\mathcal{I}$	$\|\|s_k - s_{k-1}\|\|$	$\|\|s - s_{ex}\|\|$
1	0	10^{-2}	10^{-2}	12	12.91	5.34×10^{-03}	6.80×10^{-08}	2.54×10^{-01}
			10^{-4}	17	18.30	5.34×10^{-03}	4.02×10^{-08}	2.54×10^{-01}
			0	19	16.51	5.34×10^{-03}	8.97×10^{-07}	2.54×10^{-01}
		10^{-4}	10^{-2}	21	15.83	1.11×10^{-04}	1.12×10^{-06}	2.44×10^{-02}
			10^{-4}	82	65.15	1.11×10^{-04}	8.33×10^{-06}	3.27×10^{-02}
			0	108	88.42	1.11×10^{-04}	6.71×10^{-06}	3.29×10^{-02}
		10^{-6}	10^{-2}	1	3.219	1.22×10^{-06}	0.00	0.00
			10^{-4}	49	2.003	1.22×10^{-06}	2.75×10^{-06}	5.53×10^{-04}
			0	515	225.4	1.21×10^{-06}	2.82×10^{-07}	1.57×10^{-03}
	0.2	10^{-2}	10^{-2}	10	15.89	5.34×10^{-03}	0.00	2.53×10^{-01}
			10^{-4}	18	16.40	5.34×10^{-03}	7.50×10^{-06}	2.54×10^{-01}
			0	18	16.70	5.34×10^{-03}	1.40×10^{-08}	2.54×10^{-01}
		10^{-4}	10^{-2}	97	135.1	1.12×10^{-04}	3.64×10^{-07}	3.29×10^{-02}
			10^{-4}	240	288.3	1.11×10^{-04}	5.04×10^{-06}	3.32×10^{-02}
			0	240	292.0e	1.11×10^{-04}	4.55×10^{-06}	3.32×10^{-02}
		10^{-6}	10^{-2}	3	5.804	1.84×10^{-06}	1.81×10^{-08}	6.17×10^{-02}
			10^{-4}	229	197.9	1.57×10^{-06}	4.81×10^{-06}	6.26×10^{-02}
			0	260	182.1	1.55×10^{-06}	8.40×10^{-06}	6.20×10^{-02}

Table 4.3: Results on a 64×64 grid (4096 nodes) with $\mu_1 = 1$ with $s_{ex}(t) = 1 - 8t^2(1-t)^2$. Graphs shown in Fig. B.1

μ_1	μ_2	ε_s	ε_u	k	CPU time ($\times 10^2$s)	$\mathcal{I}$	$\|\|s_k - s_{k-1}\|\|$	$\|\|s - s_{ex}\|\|$
0	0	10^{-2}	10^{-2}	16	2.243	6.86×10^{-3}	9.37×10^{-6}	3.24×10^{-1}
			10^{-4}	17	2.632	6.80×10^{-3}	6.55×10^{-6}	3.25×10^{-1}
			0	13	1.931	6.79×10^{-3}	2.24×10^{-6}	3.25×10^{-1}
		10^{-4}	10^{-2}	220	40.21	2.20×10^{-4}	1.25×10^{-6}	5.84×10^{-2}
			10^{-4}	319	58.52	1.93×10^{-4}	2.66×10^{-9}	5.74×10^{-2}
			0	243	38.93	1.92×10^{-4}	8.84×10^{-6}	5.73×10^{-2}
		10^{-6}	10^{-2}	1000	107.7	2.86×10^{-5}	2.33×10^{-4}	2.04×10^{-2}
			10^{-4}	1000	95.28	2.40×10^{-6}	1.12×10^{-4}	8.07×10^{-3}
			0	1000	94.48	2.13×10^{-6}	1.35×10^{-4}	8.00×10^{-3}
	0.2	10^{-2}	10^{-2}	16	2.288	6.86×10^{-3}	2.99×10^{-6}	3.24×10^{-1}
			10^{-4}	17	2.979	6.80×10^{-3}	7.40×10^{-6}	3.25×10^{-1}
			0	17	3.388	6.79×10^{-3}	2.44×10^{-6}	3.25×10^{-1}
		10^{-4}	10^{-2}	225	38.28	2.20×10^{-4}	5.14×10^{-6}	5.83×10^{-2}
			10^{-4}	167	26.85	1.93×10^{-4}	9.56×10^{-6}	5.69×10^{-2}
			0	303	50.51	1.92×10^{-4}	4.52×10^{-6}	5.73×10^{-2}
		10^{-6}	10^{-2}	1000	92.39	2.88×10^{-5}	3.28×10^{-4}	5.97×10^{-2}
			10^{-4}	1000	98.37	2.62×10^{-6}	5.28×10^{-4}	6.14×10^{-2}
			0	1000	95.26	2.35×10^{-6}	6.20×10^{-4}	6.15×10^{-2}

Table 4.4: Results on a 64×64 grid (4096 nodes) with $\mu_1 = 0$ with $s_{ex}(t) = 1 - 5t^3(1-t)$. Graphs shown in Fig. B.7

μ_1	μ_2	ε_s	ε_u	k	CPU time ($\times 10^2$s)	$\mathcal{I}$	$\|\|s_k - s_{k-1}\|\|$	$\|\|s - s_{ex}\|\|$
0.5	0	10^{-2}	10^{-2}	15	2.435	6.86×10^{-3}	9.77×10^{-6}	3.24×10^{-1}
			10^{-4}	16	2.806	6.80×10^{-3}	6.21×10^{-6}	3.25×10^{-1}
			0	17	3.179	6.79×10^{-3}	3.67×10^{-6}	3.25×10^{-1}
		10^{-4}	10^{-2}	52	8.314	2.20×10^{-4}	9.11×10^{-6}	5.57×10^{-2}
			10^{-4}	260	41.83	1.93×10^{-4}	7.58×10^{-6}	5.74×10^{-2}
			0	232	35.41	1.92×10^{-4}	5.89×10^{-6}	5.73×10^{-2}
		10^{-6}	10^{-2}	1000	94.18	2.86×10^{-5}	9.21×10^{-5}	1.32×10^{-2}
			10^{-4}	1000	94.08	2.39×10^{-6}	3.54×10^{-5}	6.97×10^{-3}
			0	1000	93.72	2.13×10^{-6}	4.86×10^{-5}	6.99×10^{-3}
	0.2	10^{-2}	10^{-2}	15	2.454	6.86×10^{-3}	7.49×10^{-6}	3.24×10^{-1}
			10^{-4}	15	3.294	6.80×10^{-3}	0.00	3.25×10^{-1}
			0	17	2.887	6.79×10^{-3}	7.30×10^{-6}	3.25×10^{-1}
		10^{-4}	10^{-2}	179	3.114	2.20×10^{-4}	3.66×10^{-6}	5.82×10^{-2}
			10^{-4}	177	30.25	1.93×10^{-4}	6.41×10^{-6}	5.71×10^{-2}
			0	366	56.22	1.92×10^{-4}	7.59×10^{-6}	5.73×10^{-2}
		10^{-6}	10^{-2}	1000	94.28	2.87×10^{-5}	3.38×10^{-4}	5.41×10^{-2}
			10^{-4}	1000	93.80	2.60×10^{-6}	5.74×10^{-4}	5.64×10^{-2}
			0	1000	96.53	2.32×10^{-6}	5.76×10^{-4}	5.48×10^{-2}

Table 4.5: Results on a 64×64 grid (4096 nodes) with $\mu_1 = 0.5$ with $s_{ex}(t) = 1 - 5t^3(1-t)$. Graphs shown in Fig. B.7

μ_1	μ_2	ε_s	ε_u	k	CPU time ($\times 10^2$s)	$\mathcal{I}$	$\|\|s_k - s_{k-1}\|\|$	$\|\|s - s_{ex}\|\|$
1.0	0	10^{-2}	10^{-2}	19	3.966	6.86×10^{-3}	9.69×10^{-6}	3.24×10^{-1}
			10^{-4}	18	4.312	6.80×10^{-3}	4.81×10^{-6}	3.25×10^{-1}
			0	18	2.979	6.79×10^{-3}	5.13×10^{-6}	3.25×10^{-1}
		10^{-4}	10^{-2}	136	22.73	2.20×10^{-4}	7.53×10^{-6}	5.80×10^{-2}
			10^{-4}	311	48.27	1.93×10^{-4}	0.00	5.74×10^{-2}
			0	293	45.28	1.92×10^{-4}	5.56×10^{-6}	5.73×10^{-2}
		10^{-6}	10^{-2}	1000	94.18	2.86×10^{-5}	1.88×10^{-4}	1.46×10^{-2}
			10^{-4}	1000	92.95	2.40×10^{-6}	4.77×10^{-5}	4.68×10^{-3}
			0	1000	93.18	2.13×10^{-6}	5.47×10^{-5}	4.79×10^{-3}
	0.2	10^{-2}	10^{-2}	20	4.053	6.86×10^{-3}	3.24×10^{-6}	3.24×10^{-1}
			10^{-4}	22	3.870	6.80×10^{-3}	5.38×10^{-6}	3.25×10^{-1}
			0	20	3.565	6.79×10^{-3}	6.69×10^{-6}	3.25×10^{-1}
		10^{-4}	10^{-2}	304	51.97	2.20×10^{-4}	3.22×10^{-6}	5.84×10^{-2}
			10^{-4}	337	49.27	1.93×10^{-4}	8.17×10^{-7}	5.73×10^{-2}
			0	300	43.91	1.92×10^{-4}	9.60×10^{-6}	5.73×10^{-2}
		10^{-6}	10^{-2}	1000	93.71	2.87×10^{-5}	3.13×10^{-4}	4.87×10^{-2}
			10^{-4}	1000	93.69	2.59×10^{-6}	8.28×10^{-4}	5.14×10^{-2}
			0	1000	93.22	2.32×10^{-6}	5.44×10^{-4}	5.16×10^{-2}

Table 4.6: Results on a 64×64 grid (4096 nodes) with $\mu_1 = 1$ with $s_{ex}(t) = 1 - 5t^3(1-t)$. Graphs shown in Fig. B.7

μ_1	μ_2	ε_s	ε_u	k	CPU time ($\times 10^2$s)	$\mathcal{I}$	$\|\|s_k - s_{k-1}\|\|$	$\|\|s - s_{ex}\|\|$
0	0	10^{-2}	10^{-2}	12	1.328	4.77×10^{-3}	3.82×10^{-6}	2.67×10^{-1}
			10^{-4}	12	1.433	4.70×10^{-3}	3.40×10^{-6}	2.69×10^{-1}
			0	12	1.429	4.70×10^{-3}	3.44×10^{-6}	2.69×10^{-1}
		10^{-4}	10^{-2}	379	34.72	1.24×10^{-4}	5.98×10^{-6}	5.48×10^{-2}
			10^{-4}	341	31.23	9.03×10^{-5}	9.38×10^{-6}	6.05×10^{-2}
			0	363	33.12	9.00×10^{-5}	9.45×10^{-6}	6.06×10^{-2}
		10^{-6}	10^{-2}	1000	92.41	3.30×10^{-5}	4.95×10^{-4}	1.95×10^{-2}
			10^{-4}	1000	93.03	1.32×10^{-6}	1.74×10^{-4}	3.14×10^{-2}
			0	1000	91.58	9.91×10^{-7}	2.01×10^{-4}	3.15×10^{-2}
	0.2	10^{-2}	10^{-2}	16	2.116	4.77×10^{-3}	1.80×10^{-6}	2.67×10^{-1}
			10^{-4}	16	2.054	4.70×10^{-3}	1.82×10^{-6}	2.69×10^{-1}
			0	16	2.318	4.70×10^{-3}	1.82×10^{-6}	2.69×10^{-1}
		10^{-4}	10^{-2}	521	50.90	1.24×10^{-4}	9.73×10^{-6}	5.48×10^{-2}
			10^{-4}	463	45.42	9.03×10^{-5}	9.89×10^{-6}	6.05×10^{-2}
			0	494	49.02	9.00×10^{-5}	8.19×10^{-6}	6.06×10^{-2}
		10^{-6}	10^{-2}	1000	95.51	3.32×10^{-5}	3.54×10^{-4}	7.96×10^{-2}
			10^{-4}	1000	98.59	1.68×10^{-6}	1.80×10^{-4}	7.35×10^{-2}
			0	1000	100.1	1.36×10^{-6}	5.01×10^{-4}	7.35×10^{-2}

Table 4.7: Results on a 64×64 grid (4096 nodes) with $\mu_1 = 0$ with $s_{ex}(t) = 1 - 8t^2(1-t)^2$. Graphs shown in Fig. B.13

μ_1	μ_2	ε_s	ε_u	k	CPU time ($\times 10^2$s)	$\mathcal{I}$	$\|\|s_k - s_{k-1}\|\|$	$\|\|s - s_{ex}\|\|$
0.5	0	10^{-2}	10^{-2}	18	2.212	4.77×10^{-3}	6.17×10^{-6}	2.67×10^{-1}
			10^{-4}	18	2.097	4.70×10^{-3}	6.16×10^{-6}	2.69×10^{-1}
			0	18	2.243	4.70×10^{-3}	5.85×10^{-6}	2.69×10^{-1}
		10^{-4}	10^{-2}	437	39.46	1.24×10^{-4}	9.66×10^{-6}	5.48×10^{-2}
			10^{-4}	415	37.65	9.03×10^{-5}	8.58×10^{-6}	6.04×10^{-2}
			0	408	37.17	9.00×10^{-5}	9.72×10^{-6}	6.05×10^{-2}
		10^{-6}	10^{-2}	1000	90.74	3.29×10^{-5}	4.06×10^{-4}	9.97×10^{-3}
			10^{-4}	1000	90.72	1.30×10^{-6}	5.55×10^{-5}	1.18×10^{-2}
			0	1000	90.28	9.74×10^{-7}	5.65×10^{-5}	1.20×10^{-2}
	0.2	10^{-2}	10^{-2}	12	1.497	4.77×10^{-3}	7.18×10^{-6}	2.67×10^{-1}
			10^{-4}	13	1.563	4.70×10^{-3}	8.88×10^{-6}	2.69×10^{-1}
			0	13	1.568	4.70×10^{-3}	9.04×10^{-6}	2.69×10^{-1}
		10^{-4}	10^{-2}	506	48.81	1.24×10^{-4}	9.82×10^{-6}	5.48×10^{-2}
			10^{-4}	482	47.78	9.03×10^{-5}	8.69×10^{-6}	6.05×10^{-2}
			0	480	46.35	9.00×10^{-5}	8.53×10^{-6}	6.05×10^{-2}
		10^{-6}	10^{-2}	1000	94.93	3.31×10^{-5}	5.27×10^{-4}	5.83×10^{-2}
			10^{-4}	1000	95.04	1.58×10^{-6}	3.52×10^{-4}	5.44×10^{-2}
			0	1000	94.95	1.26×10^{-6}	5.35×10^{-4}	5.44×10^{-2}

Table 4.8: Results on a 64×64 grid (4096 nodes) with $\mu_1 = 0.5$ with $s_{ex}(t) = (1+|2t-1|)/2$. Graphs shown in Fig. B.13

μ_1	μ_2	ε_s	ε_u	k	CPU time ($\times 10^2$s)	$\mathcal{I}$	$\|\|s_k - s_{k-1}\|\|$	$\|\|s - s_{ex}\|\|$
1.0	0	10^{-2}	10^{-2}	15	1.883	4.77×10^{-3}	6.79×10^{-6}	2.67×10^{-1}
			10^{-4}	15	1.892	4.70×10^{-3}	6.40×10^{-6}	2.69×10^{-1}
			0	15	1.889	4.70×10^{-3}	6.32×10^{-6}	2.69×10^{-1}
		10^{-4}	10^{-2}	486	44.19	1.24×10^{-4}	9.90×10^{-6}	5.48×10^{-2}
			10^{-4}	453	40.99	9.03×10^{-5}	8.77×10^{-6}	6.04×10^{-2}
			0	473	42.83	9.00×10^{-5}	9.36×10^{-6}	6.05×10^{-2}
		10^{-6}	10^{-2}	1000	91.15	3.28×10^{-5}	2.48×10^{-4}	1.35×10^{-2}
			10^{-4}	1000	90.73	1.31×10^{-6}	1.18×10^{-4}	3.45×10^{-3}
			0	1000	90.31	9.89×10^{-7}	7.83×10^{-5}	3.64×10^{-3}
	0.2	10^{-2}	10^{-2}	16	2.572	4.77×10^{-3}	5.77×10^{-6}	2.67×10^{-1}
			10^{-4}	19	2.713	4.70×10^{-3}	7.11×10^{-6}	2.69×10^{-1}
			0	19	2.772	4.70×10^{-3}	7.14×10^{-6}	2.69×10^{-1}
		10^{-4}	10^{-2}	511	48.90	1.24×10^{-4}	9.97×10^{-6}	5.48×10^{-2}
			10^{-4}	479	47.27	9.03×10^{-5}	8.90×10^{-6}	6.05×10^{-2}
			0	484	47.34	9.00×10^{-5}	8.83×10^{-6}	6.05×10^{-2}
		10^{-6}	10^{-2}	1000	94.87	3.30×10^{-5}	3.79×10^{-4}	6.06×10^{-2}
			10^{-4}	1000	94.78	1.55×10^{-6}	3.92×10^{-4}	4.64×10^{-2}
			0	1000	95.03	1.23×10^{-6}	5.45×10^{-4}	4.63×10^{-2}

Table 4.9: Results on a 64×64 grid (4096 nodes) with $\mu_1 = 1$ with $s_{ex}(t) = (1 + |2t - 1|)/2$. Graphs shown in Fig. B.13

4.7 Conclusions

We have developed a method to identify an unknown boundary given an elliptic PDE with boundary data to fit. This was done by reformulating the free boundary problem as a problem to find a map $s(t)$ whose graph is the unknown boundary. A cost function was developed and minimised via a gradient descent method. The construction of a test-case scenario has shown that the method outlined here can convincingly converge to a constructed solution. This is dependent upon the choice of the regularisation parameter ε_s and the stability parameter ε_u, since the larger these parameters are, the further we move away from the constructed solution as we necessarily alter the problem. However, that the final solutions can come within 10^{-2} and even less of the constructed solution (measured in the $L2$-norm) is encouraging.

To build upon this methodology we may consider a number of things:

- replacing Laplace's equation with an equation of flow, such as time-independent Stokes flow, to simulate blood flow
- since the real system is transient in nature, time dependence can be introduced.
- in this chapter, the geometry we have worked with is suitable for an axially symmetric assumption of the vessel. Since arteries do not possess such symmetry in real life, this assumption can clearly be improved upon. Since 3-D ultrasound data can be obtained, further work could consider 3-D geometries.

With these further elements a more complete model can be obtained.

Chapter 5

Conclusion and further work

In this thesis we have considered atherosclerosis and the mathematical modelling of it. Due to the high prevalence and the number of deaths associated with atherosclerosis, it is paramount to be able to provide effective treatment to individuals. Mathematical modelling has a role in this endeavour: in its ability to provide safe, cost-effective and highly reproducible simulations, it furnishes an addition to clinical trials and experiments to augment insights on the disease. We have examined the necessary tools to model reaction-diffusion systems and also looked at a shape identification problem. The work laid out in this thesis can be extended in a number of ways, some of which have already been suggested in previous chapters. A key element of atherosclerosis that we have not presented here is blood flow. This is influential on the progression on the disease being the supplier of key participants such as white blood cells and lipids to the vessel wall and bearing physical forces on the lesion. The addition of this element can be coupled to the ones presented in this thesis to provide a more accurate description of the system in real life.

As the trend for patient-specific modelling continues, the need for better mathematical models becomes more essential. The efficacy of such models is tied to the availability of its parameters, i.e. how easy it is to get the parameters needed for each individual. Since the parameters required are not always easy to obtain, inverse modelling has its role and we have seen an example of one in this thesis.

One of the main questions remains: given a plaque, how likely is it to rupture and what is the best course of treatment? This is the question most relevant to clinicians who treat patients face-to-face. This avenue of research is still very much being explored and new

studies are being conducted, such as the second European Carotid Surgery Trial (ECST-2). Success of mathematical modelling can be seen in the related area of abdominal aortic aneurysms, where predicting aneurysm rupture is key. For this, commercial software exists (*A4clinics* from *VASCOPS*) which makes use of finite element modelling with patient specific CT data to provide clinicians with risk of stroke index and suggests courses of treatment. This was built upon research in the literature (Geest et al. 2006, Gasser et al. 2008, Auer & Gasser 2010, Gasser et al. 2010) and concentrates on accurate constitutive modelling of the aneurysm wall using the equations of solid mechanics. In the future then, perhaps a similar tool can be built in the area of atherosclerosis where its benefits may help to save lives.

Appendices

Appendix A

A.1 Exploring the Turing Space

In this appendix we shall use the numerical methods presented in Chapter 2 to test agreement with the linear stability analysis in (2.1). Recall, that for the reaction diffusion equation

$$\begin{cases} \dfrac{\partial u}{\partial t} - \nabla^2 u = \gamma f(u,v), \\ \dfrac{\partial v}{\partial t} - d\nabla^2 v = \gamma g(u,v), \qquad \text{in } \Omega \times I, \end{cases} \tag{A.1.1}$$

for some unknown variables $u(\boldsymbol{x},t)$, $v(\boldsymbol{x},t)$, $\boldsymbol{x}\in\Omega$ for some domain Ω, $t \in I$ for some time interval I, some given functions f and g which depend on the unknown variables and some $d > 0$, we have a Turing instability if the following four conditions hold:

$$f_u + g_v < 0 \tag{A.1.2}$$

$$f_u g_v - f_v g_u > 0 \tag{A.1.3}$$

$$df_u + g_v > 0 \tag{A.1.4}$$

$$(df_u + g_v)^2 - 4d(f_u g_v - f_v g_u) > 0, \tag{A.1.5}$$

where the derivatives are evaluated at the equilibrium values. Now, for the Schnakenberg equation defined by

$$\begin{aligned} f(u,v) &= a - u + u^2 v \\ g(u,v) &= b - u^2, \end{aligned} \tag{A.1.6}$$

for some $a, b > 0$, it is straightforward to calculate the derivates at the equilibrium values (2.1.2), so that equations (A.1.2)-(A.1.5) can be written

$$b - a - (a+b)^3 < 0 \tag{A.1.7}$$

$$(a+b)^2 > 0 \tag{A.1.8}$$

$$d(b-a) - (a+b)^3 > 0 \tag{A.1.9}$$

$$\left(d(b-a) - (a+b)^3\right)^2 - 4d(a+b)^4 > 0. \tag{A.1.10}$$

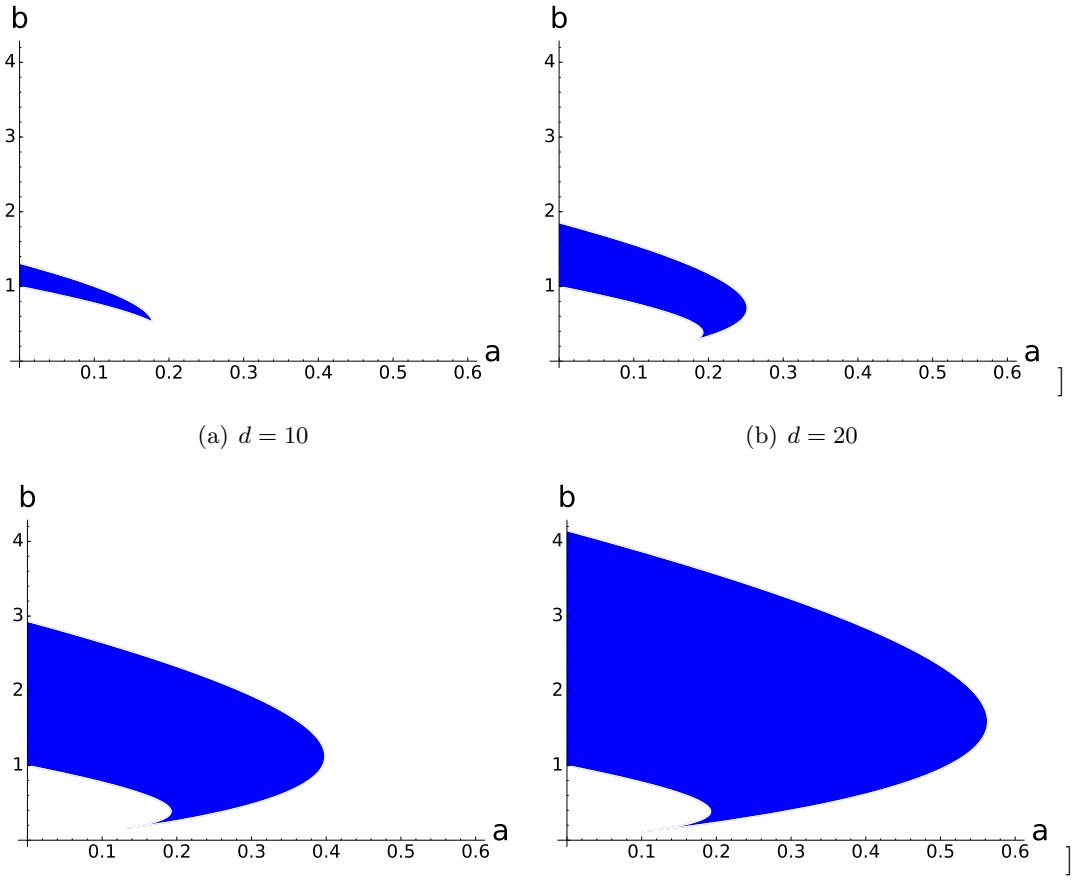
The four conditions (A.1.7)-(A.1.10) define the Turing space. The four conditions make it clear that the Turing space depends on the parameters a, b and d only. We note that since a and b are real and positive, condition A.1.8 will always be true. The Turing space for different fixed values of d are shown in Fig. A.1 as a function of a and b. The Turing space shows for which values of the parameters a, b and d a Turing instability may occur. The wavenumbers which are excited are governed by the inequality (2.1.10). In particular, this relation is governed by the parameter γ. To ensure a Turing instability, γ must be large enough to include some wavenumbers in inequality (2.1.10).

To see if our numerical methods presented in Chapter 2 agree with this analysis, let us perform some simulations here. Let us fix the parameter $d = 50$. Let us also fix $a = 0.1$ and vary b using the values $(b_0, b_1, b_2, b_3) = (0.1, 1, 2, 3)$. Then from Fig. A.1, for $d = 50$ and $a = 0.1$, we can see that the values b_0 and b_3 lie outside of the Turing space while the values b_1 and b_2 lie within. Therefore, one would expect that for γ large enough, we would see patterns arising using the b_1 and b_2, whilst we should not see any patterns for b_0 and b_3.

Let us take the domain Ω to be the unit circle. We solve the Schnakenberg system using the fractional-step θ scheme with the Newton method presented in Section 2.2 using a mesh comprised of 1028 cells shown in Fig. A.2. The final solution u of the numerical simulations using the time step 10^{-3} and $\gamma = 29$ are shown in Fig. A.3. As expected, we see patterns for b_1 and b_2, whilst for b_0 and b_3 the solution tends to the steady state value

$$u_{eq} = a + b \tag{A.1.11}$$

Thus, the numerical methods presented in Chapter 2 seem to be able to reflect the predictions made by the linear stability analysis. This provides further confidence that the methods presented result in accurate numerical results.



(a) $d = 10$ (b) $d = 20$

(c) $d = 50$ (d) $d = 100$

Figure A.1: The Turing space for the Schakenberg equation for different values of d. The coloured area shows the region where all the conditions (A.1.7)-(A.1.10) are all satisfied.

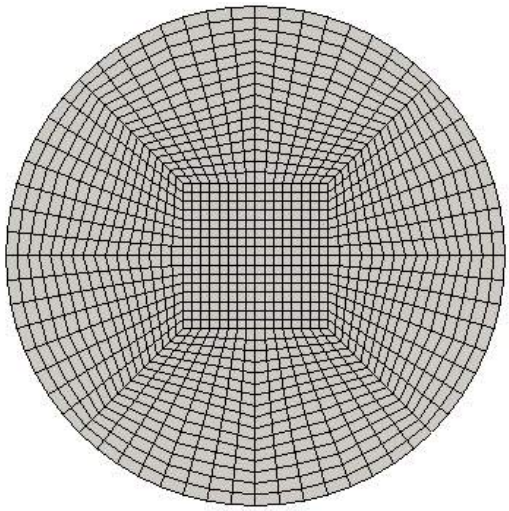
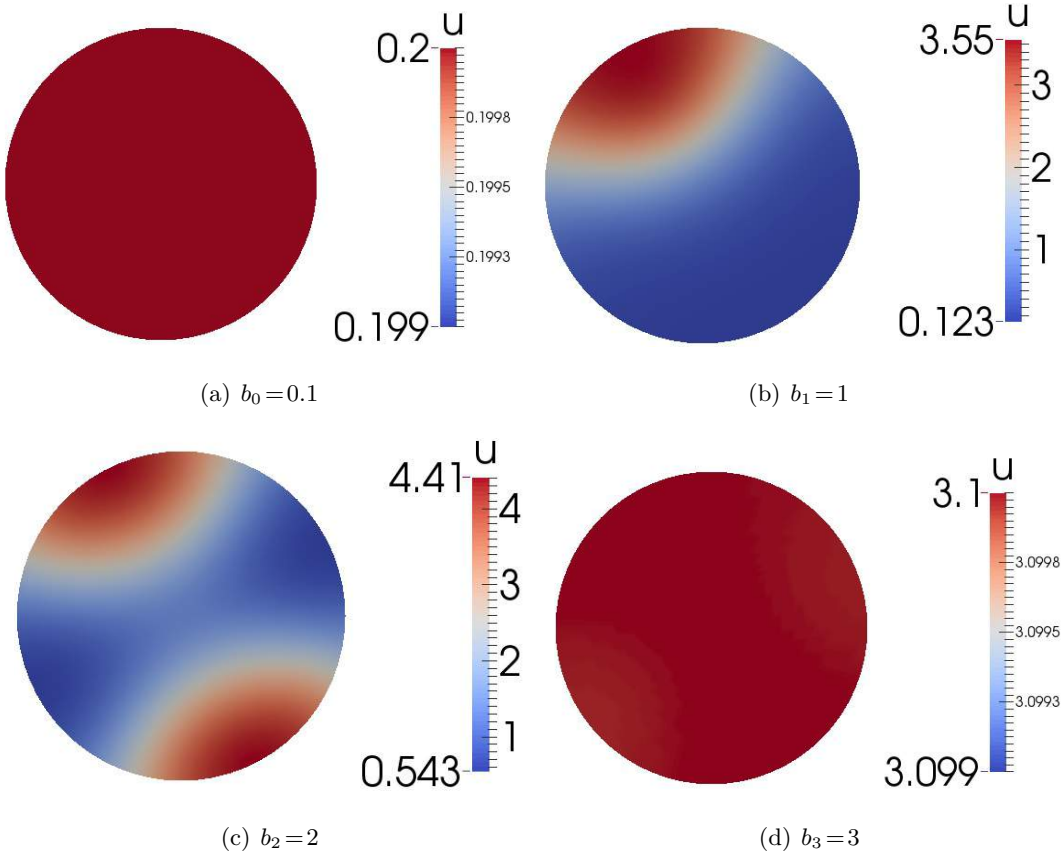


Figure A.2: The mesh used to calculate the numerical results shown in Fig. A.3.



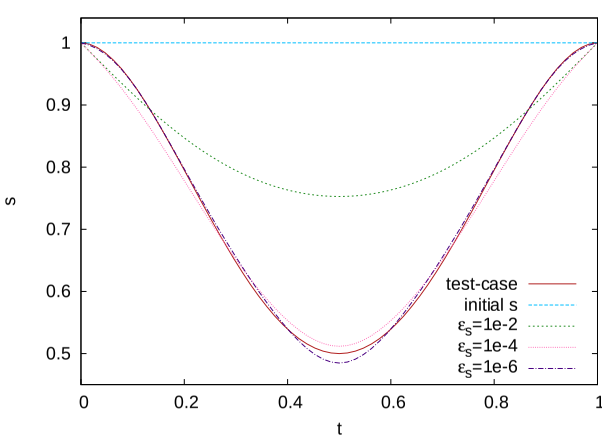
(a) $b_0 = 0.1$ (b) $b_1 = 1$

(c) $b_2 = 2$ (d) $b_3 = 3$

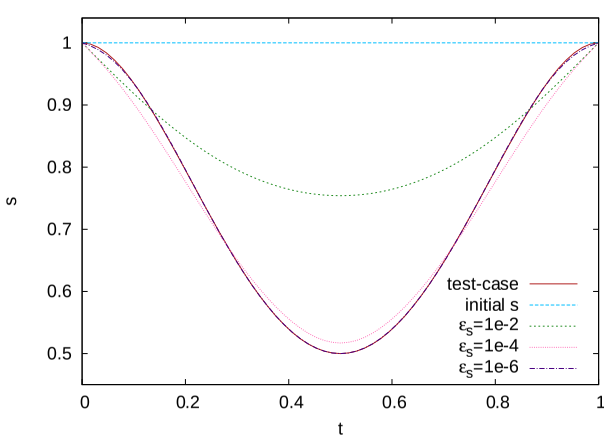
Figure A.3: Numerical solutions of the Schnakenberg system with $d = 50$, $a = 0.1$ and $\gamma = 29$ for diffrent values of b.

Appendix B

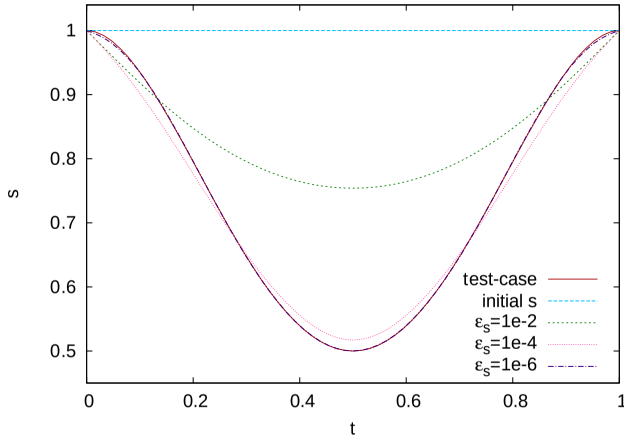
What follows is the full list figures of the numerical results obtained for Chapter 4. See Section 4.6 for details.



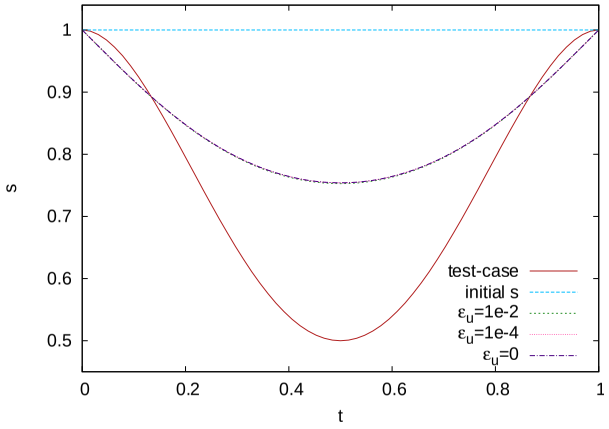
(a) $\varepsilon_u = 10^{-2}$



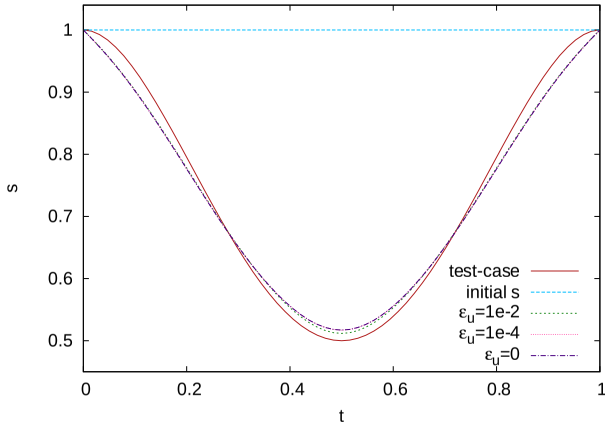
(b) $\varepsilon_u = 10^{-4}$



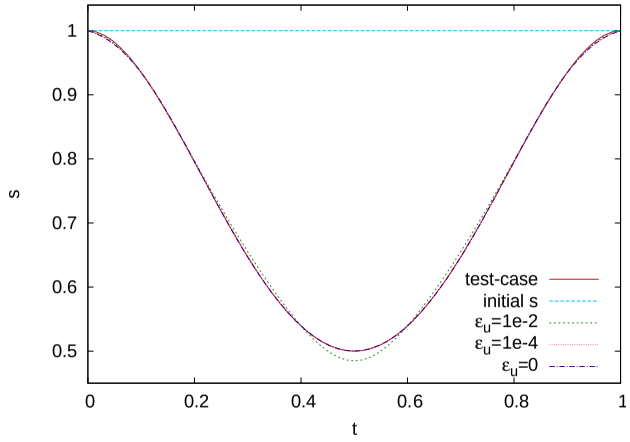
(c) $\varepsilon_u = 0$



(d) $\varepsilon_s = 10^{-2}$

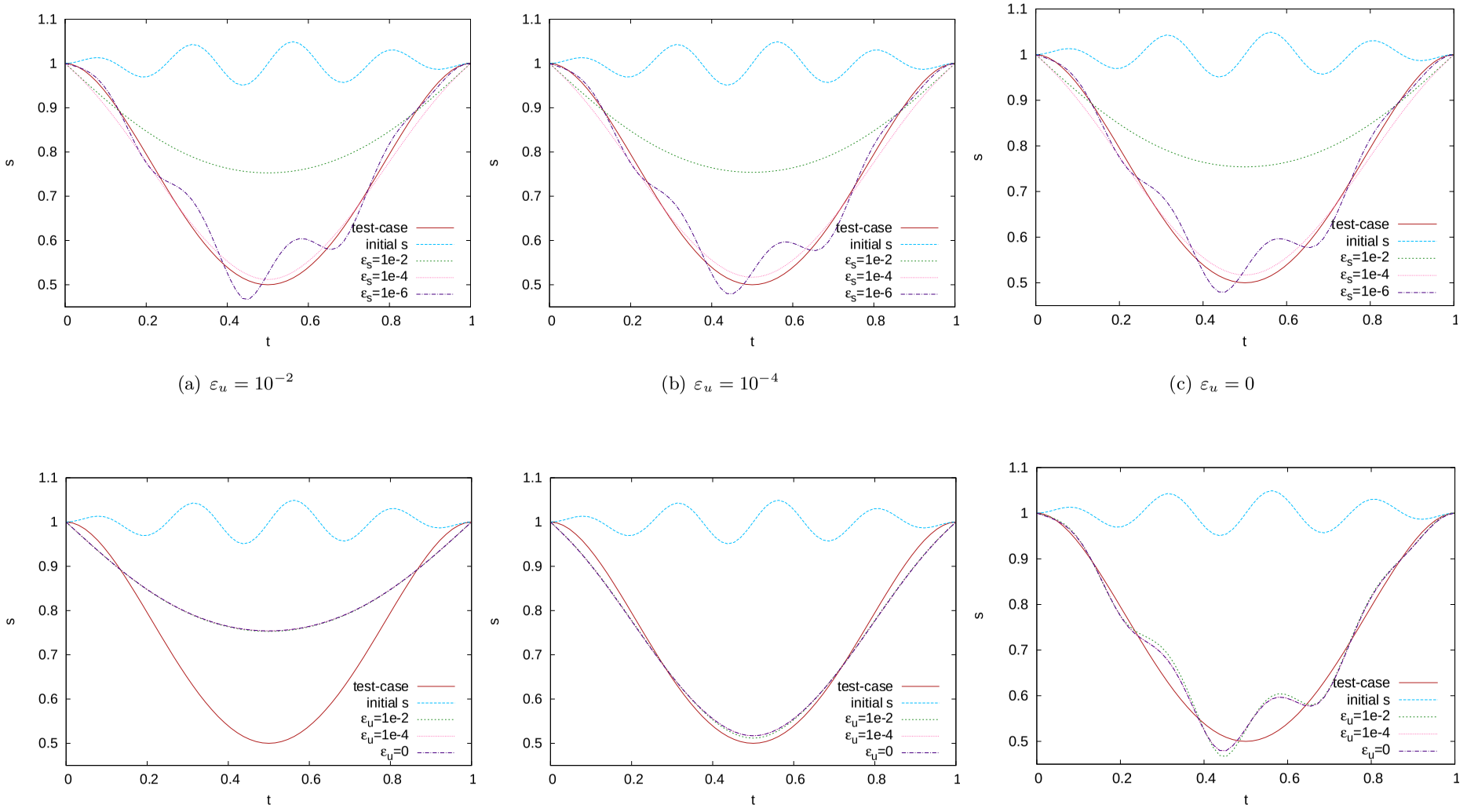


(e) $\varepsilon_s = 10^{-4}$



(f) $\varepsilon_s = 10^{-6}$

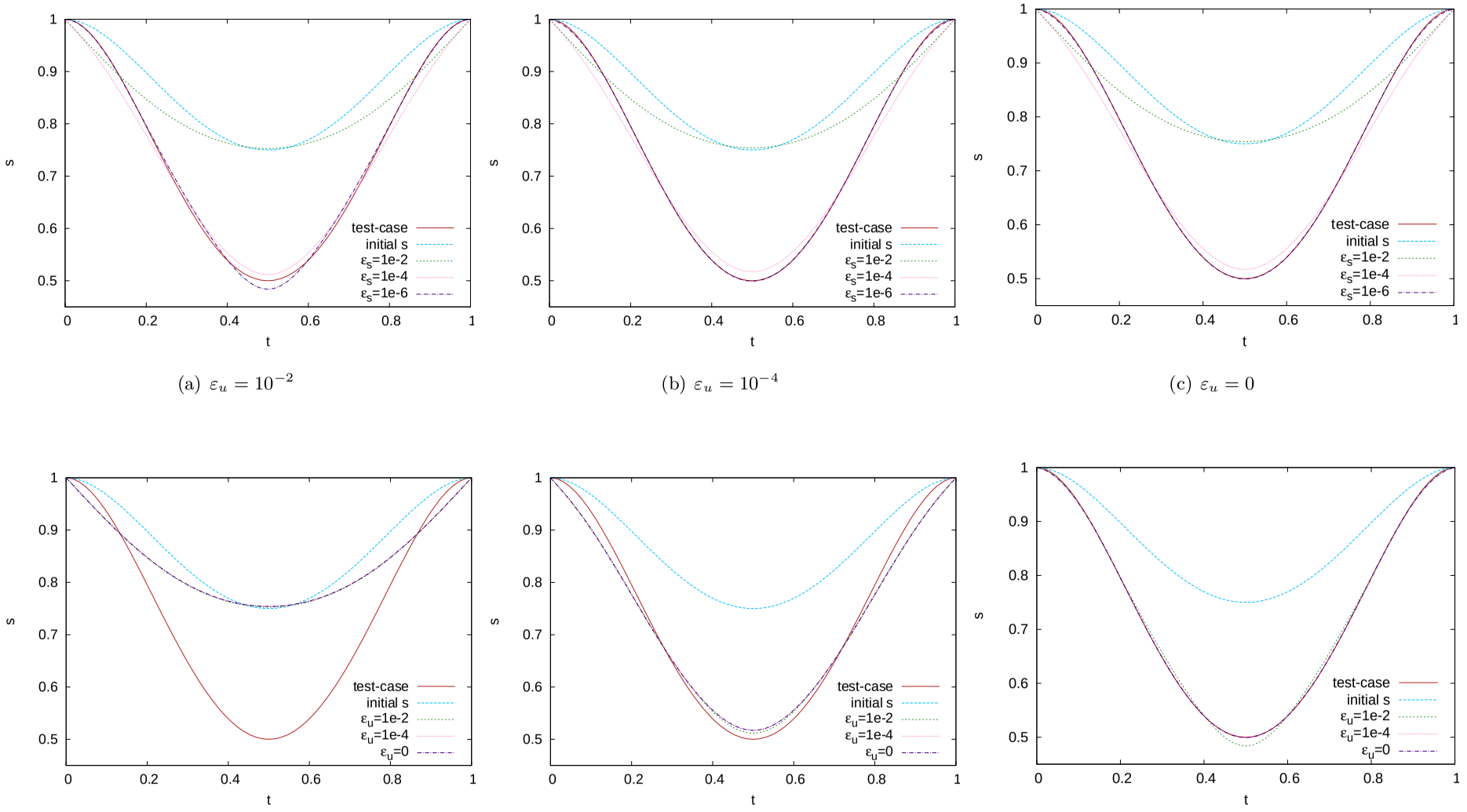
Figure B.1: Results for $s_0(t) = 1$ and $s_{ex}(t) = 1 - 8t^2(1-t)^2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.1.



(a) $\varepsilon_u = 10^{-2}$ (b) $\varepsilon_u = 10^{-4}$ (c) $\varepsilon_u = 0$

(d) $\varepsilon_s = 10^{-2}$ (e) $\varepsilon_s = 10^{-4}$ (f) $\varepsilon_s = 10^{-6}$

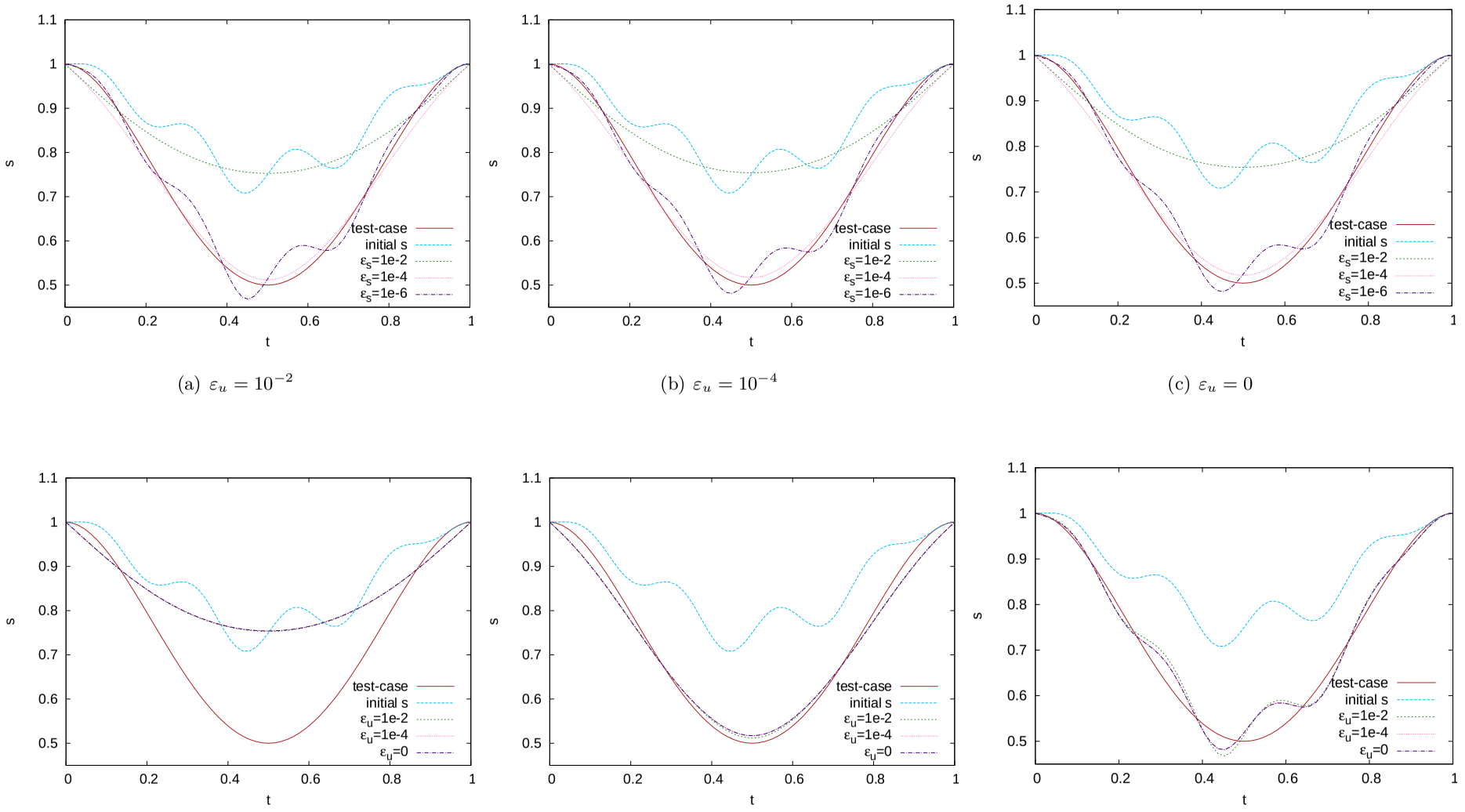
Figure B.2: Results for $s_0(t) = 1 + 0.2t(1-t)\sin(8\pi t)$ and $s_{ex}(t) = 1 - 8t^2(1-t)^2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.1.



(a) $\varepsilon_u = 10^{-2}$ (b) $\varepsilon_u = 10^{-4}$ (c) $\varepsilon_u = 0$

(d) $\varepsilon_s = 10^{-2}$ (e) $\varepsilon_s = 10^{-4}$ (f) $\varepsilon_s = 10^{-6}$

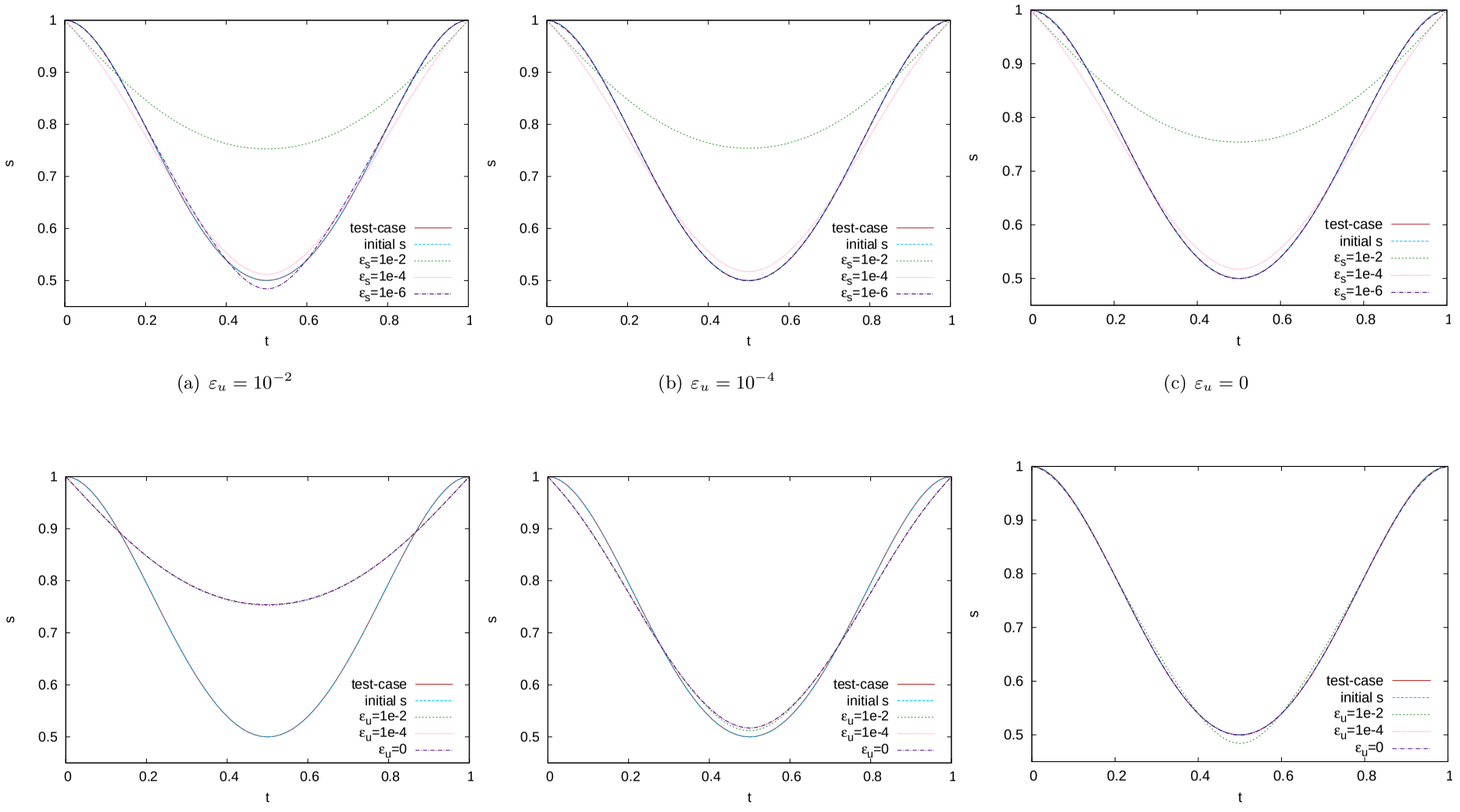
Figure B.3: Results for $s_0(t) = 0.5 + 0.5 s_{ex}(t)$ and $s_{ex}(t) = 1 - 8t^2(1-t)^2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.2.



(a) $\varepsilon_u = 10^{-2}$ (b) $\varepsilon_u = 10^{-4}$ (c) $\varepsilon_u = 0$

(d) $\varepsilon_s = 10^{-2}$ (e) $\varepsilon_s = 10^{-4}$ (f) $\varepsilon_s = 10^{-6}$

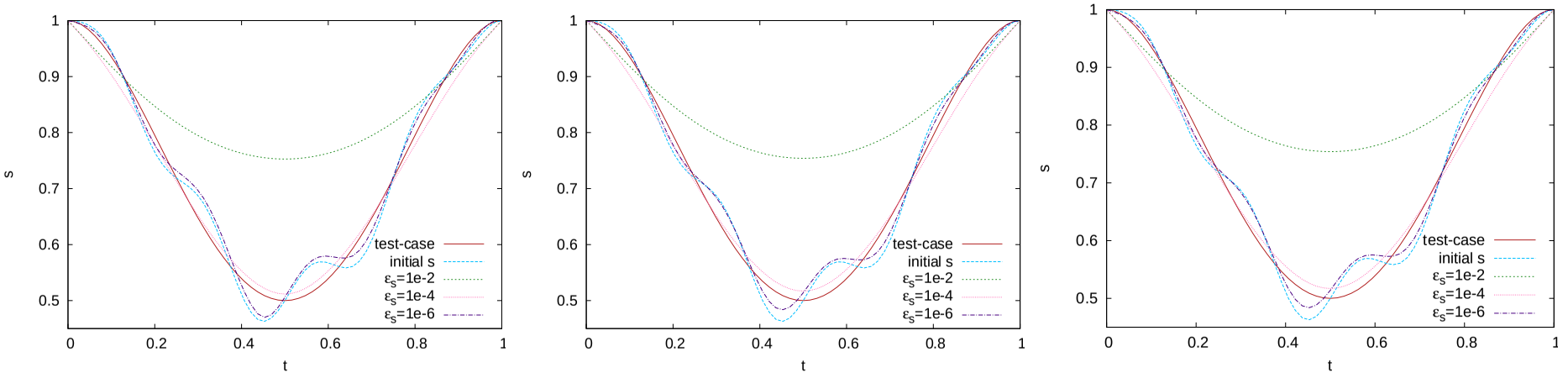
Figure B.4: Results for $s_0(t) = 0.5 + 0.5 s_{ex}(t) + 0.2t(1-t)\sin(8pi * t)$ and $s_{ex}(t) = 1 - 8t^2(1-t)^2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.2.



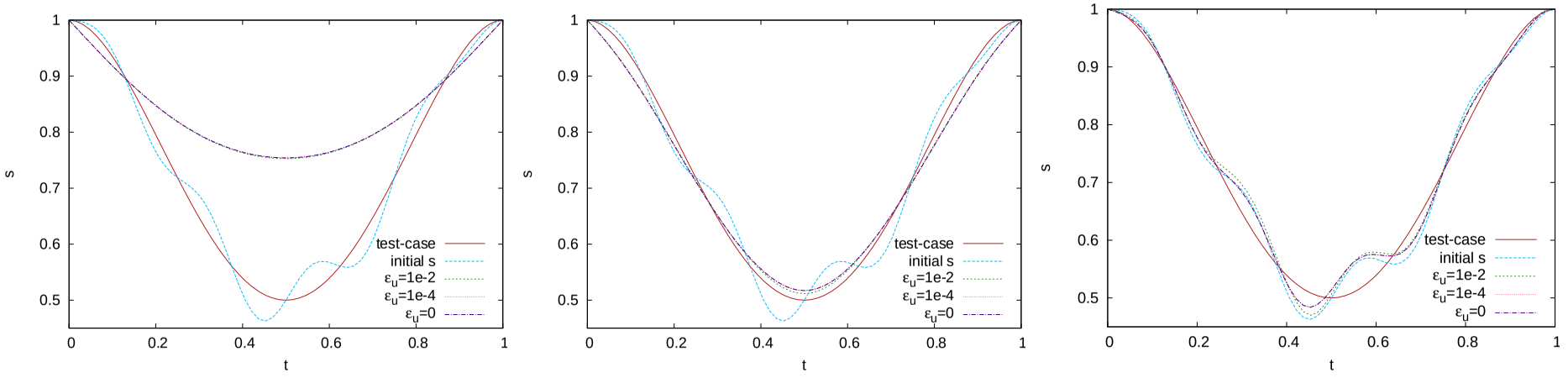
(a) $\varepsilon_u = 10^{-2}$ (b) $\varepsilon_u = 10^{-4}$ (c) $\varepsilon_u = 0$

(d) $\varepsilon_s = 10^{-2}$ (e) $\varepsilon_s = 10^{-4}$ (f) $\varepsilon_s = 10^{-6}$

Figure B.5: Results for $s_0(t) = s_{ex}(t)$ and $s_{ex}(t) = 1 - 8t^2(1-t)^2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.3.

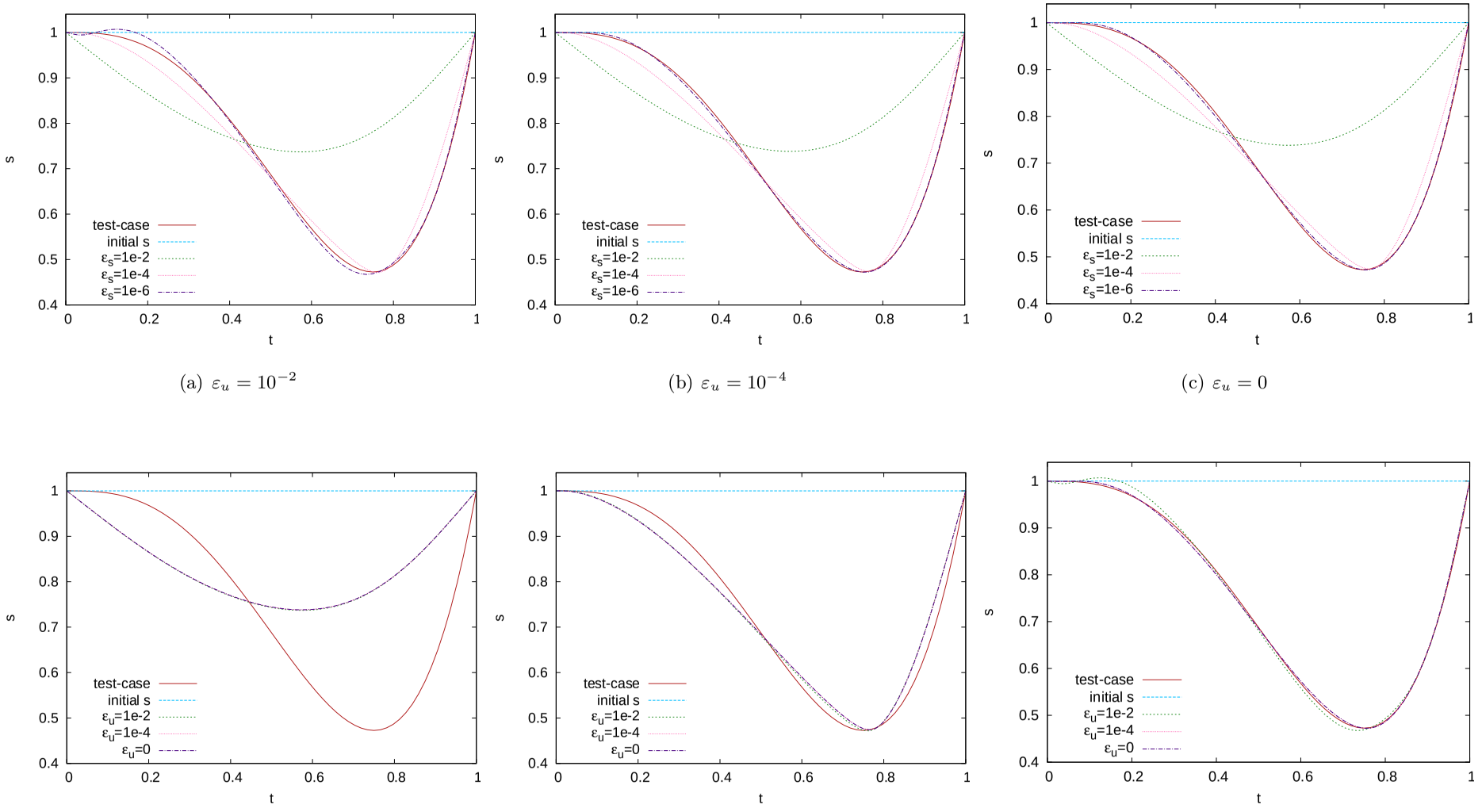


(a) $\varepsilon_u = 10^{-2}$ (b) $\varepsilon_u = 10^{-4}$ (c) $\varepsilon_u = 0$



(d) $\varepsilon_s = 10^{-2}$ (e) $\varepsilon_s = 10^{-4}$ (f) $\varepsilon_s = 10^{-6}$

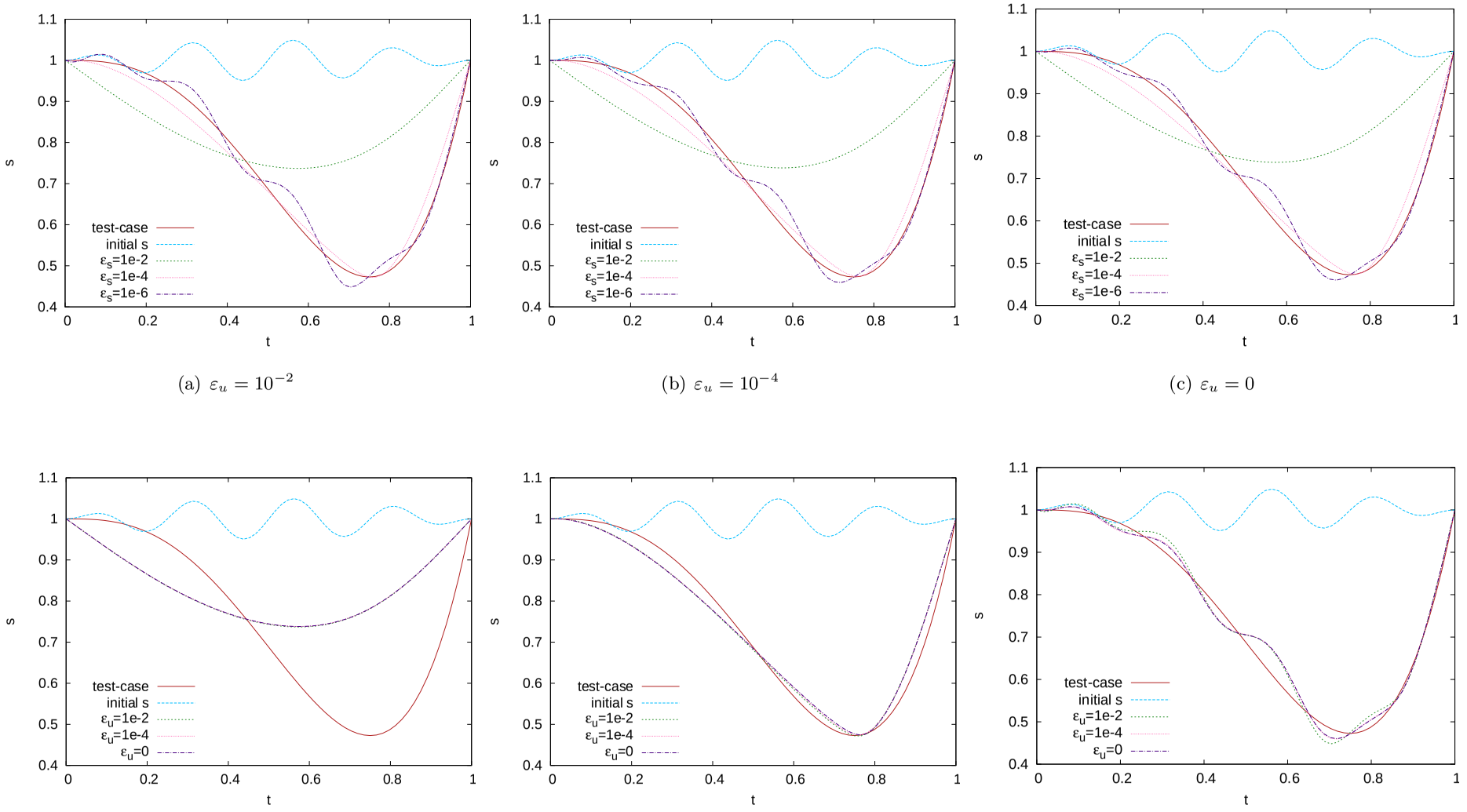
Figure B.6: Results for $s_0(t) = s_{ex}(t) + 0.2t(1-t)\sin(8\pi t)$ and $s_{ex}(t) = 1 - 8t^2(1-t)^2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.3.



(a) $\varepsilon_u = 10^{-2}$ (b) $\varepsilon_u = 10^{-4}$ (c) $\varepsilon_u = 0$

(d) $\varepsilon_s = 10^{-2}$ (e) $\varepsilon_s = 10^{-4}$ (f) $\varepsilon_s = 10^{-6}$

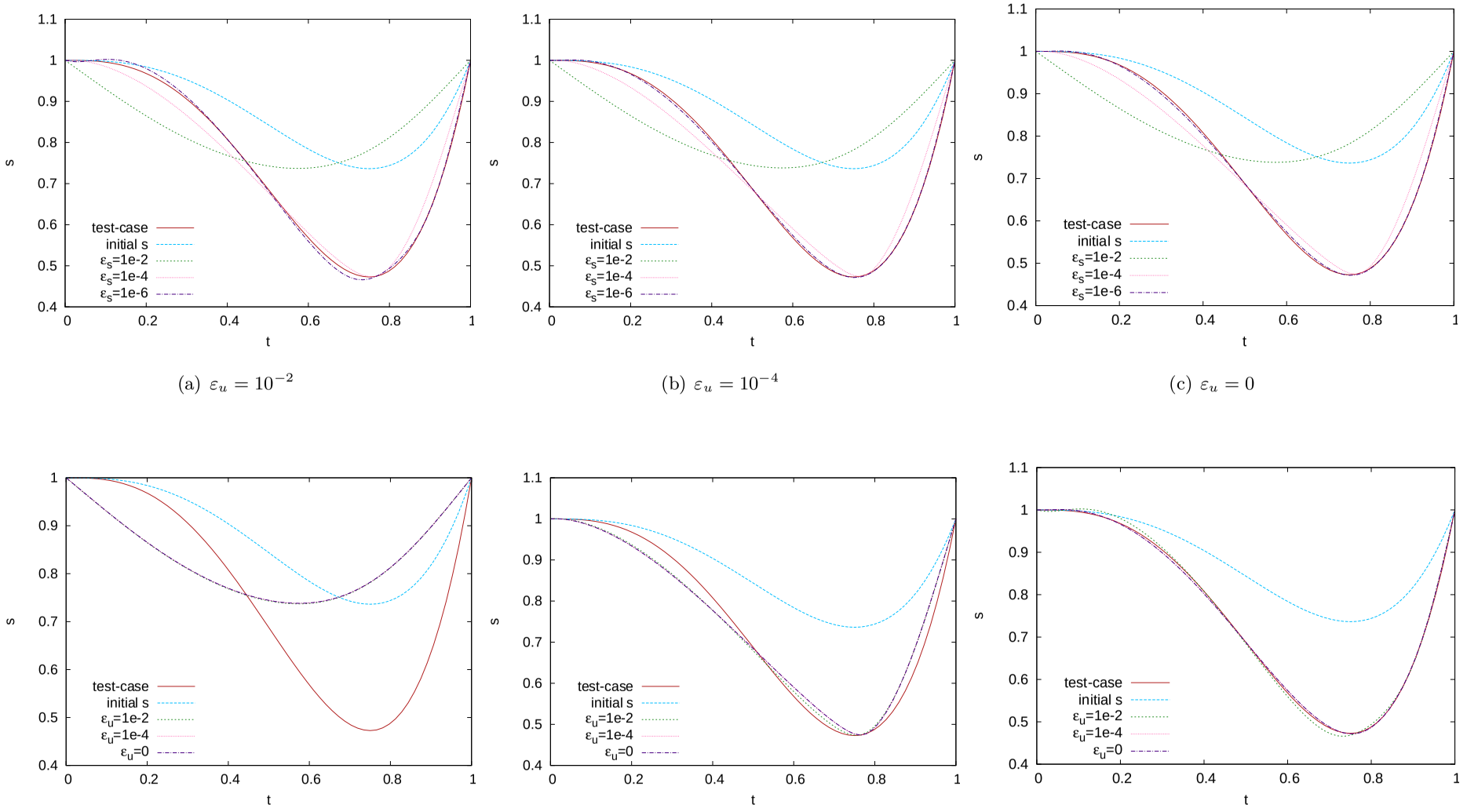
Figure B.7: Results for $s_0(t) = 1$ and $s_{ex}(t) = 1 - 5t^3(1-t)$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.4.



(a) $\varepsilon_u = 10^{-2}$ (b) $\varepsilon_u = 10^{-4}$ (c) $\varepsilon_u = 0$

(d) $\varepsilon_s = 10^{-2}$ (e) $\varepsilon_s = 10^{-4}$ (f) $\varepsilon_s = 10^{-6}$

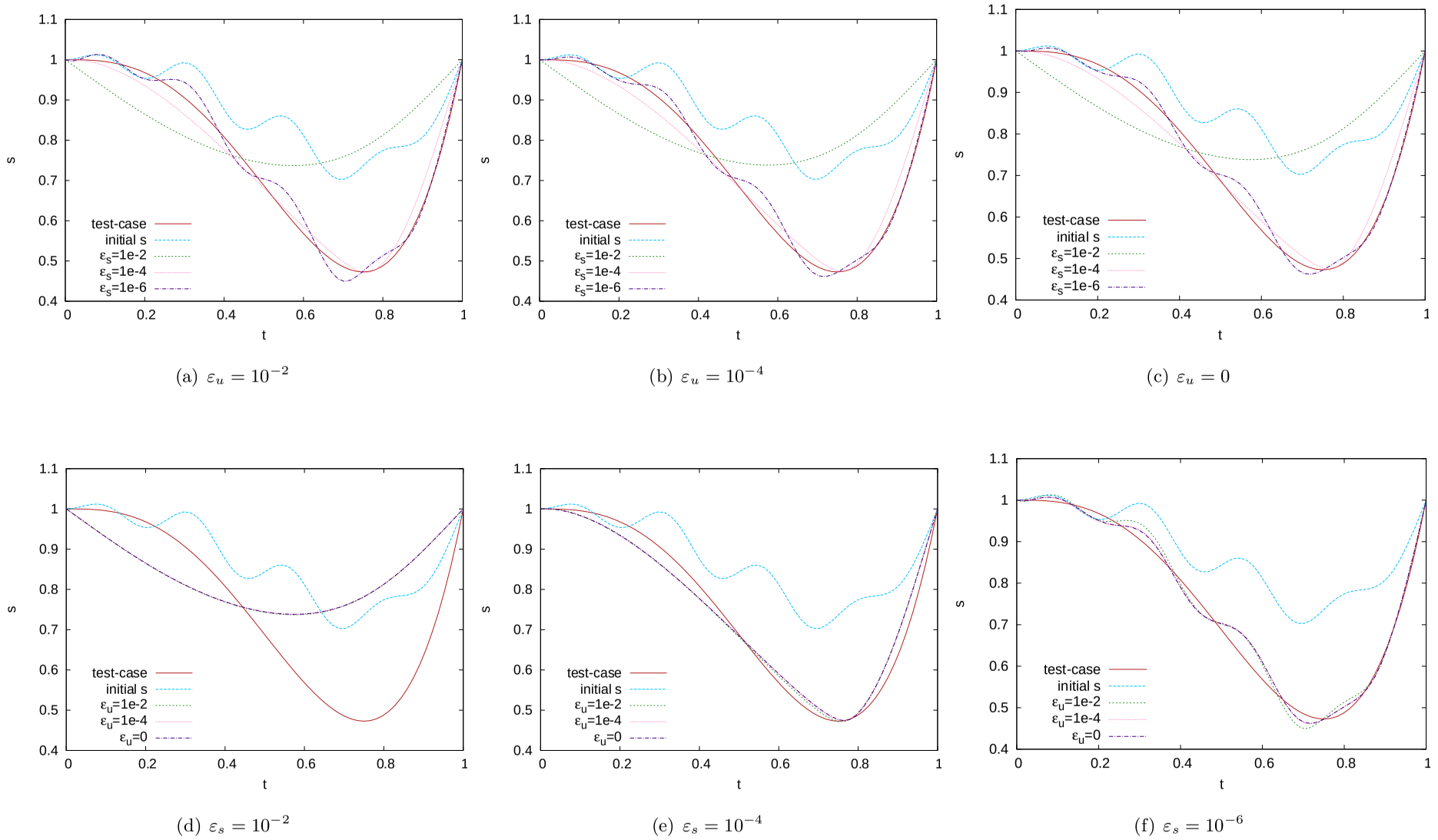
Figure B.8: Results for $s_0(t) = 1 + 0.2t(1-t)\sin(8\pi t)$ and $s_{ex}(t) = 1 - 5t^3(1-t)$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.4.



(a) $\varepsilon_u = 10^{-2}$ (b) $\varepsilon_u = 10^{-4}$ (c) $\varepsilon_u = 0$

(d) $\varepsilon_s = 10^{-2}$ (e) $\varepsilon_s = 10^{-4}$ (f) $\varepsilon_s = 10^{-6}$

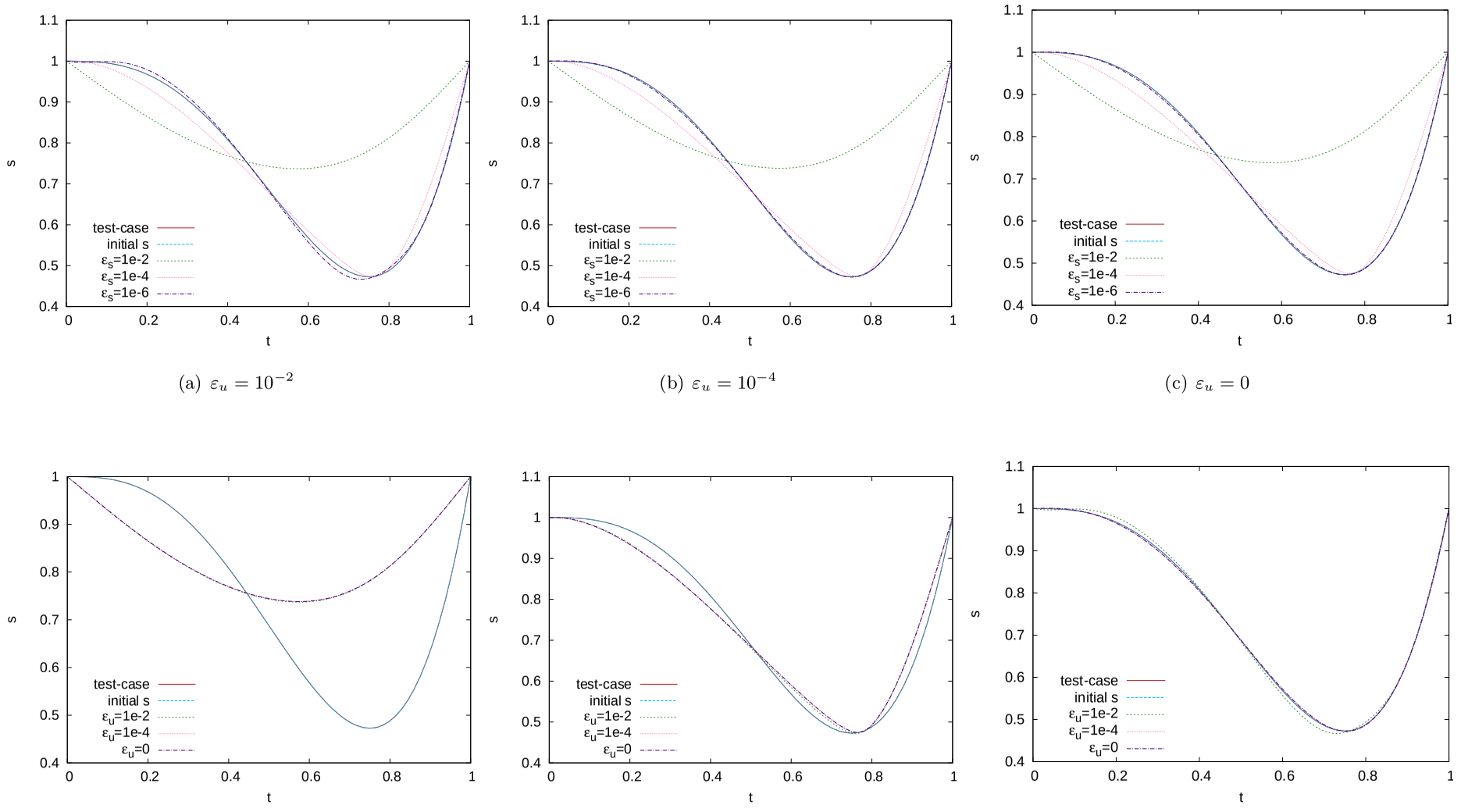
Figure B.9: Results for $s_0(t) = 0.5 + 0.5 s_{ex}(t)$ and $s_{ex}(t) = 1 - 5t^3(1-t)$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.5.



(a) $\varepsilon_u = 10^{-2}$ (b) $\varepsilon_u = 10^{-4}$ (c) $\varepsilon_u = 0$

(d) $\varepsilon_s = 10^{-2}$ (e) $\varepsilon_s = 10^{-4}$ (f) $\varepsilon_s = 10^{-6}$

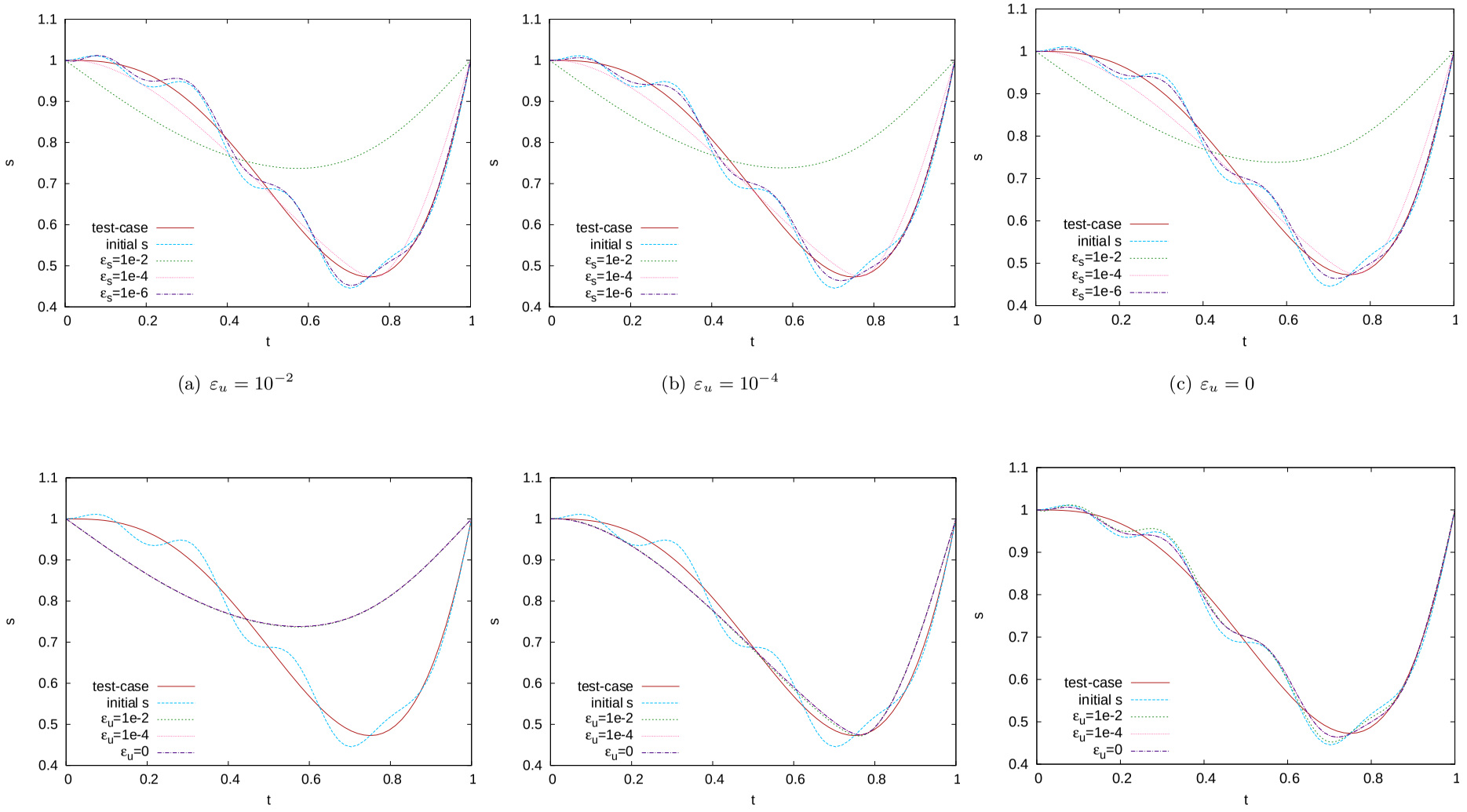
Figure B.10: Results for $s_0(t) = 0.5 + 0.5s_{ex}(t) + 0.2t(1-t)\sin(8pi * t)$ and $s_{ex}(t) = 1 - 5t^3(1-t)$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.5.



(a) $\varepsilon_u = 10^{-2}$ (b) $\varepsilon_u = 10^{-4}$ (c) $\varepsilon_u = 0$

(d) $\varepsilon_s = 10^{-2}$ (e) $\varepsilon_s = 10^{-4}$ (f) $\varepsilon_s = 10^{-6}$

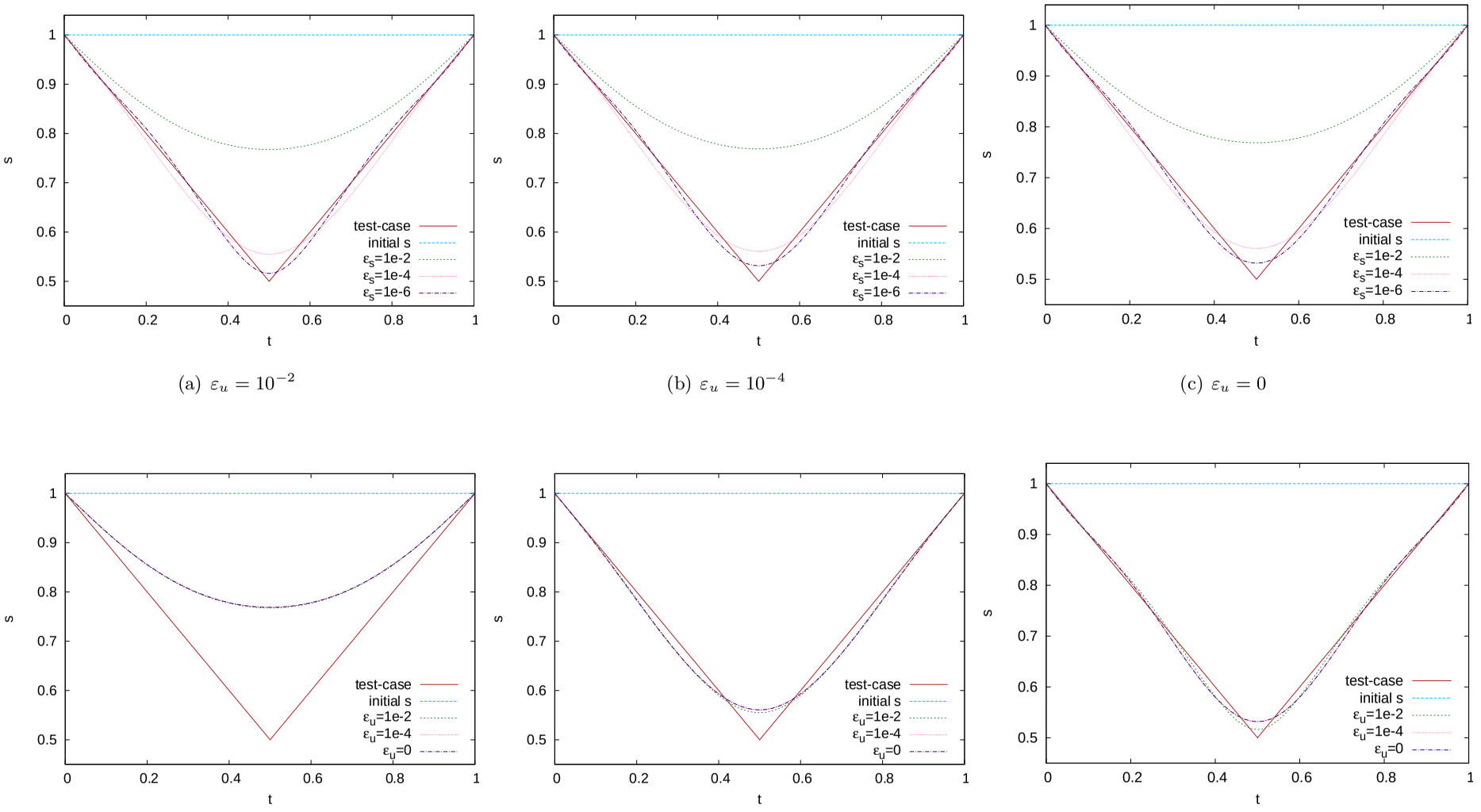
Figure B.11: Results for $s_0(t) = s_{ex}(t)$ and $s_{ex}(t) = 1 - 5t^3(1-t)$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.6.



(a) $\varepsilon_u = 10^{-2}$ (b) $\varepsilon_u = 10^{-4}$ (c) $\varepsilon_u = 0$

(d) $\varepsilon_s = 10^{-2}$ (e) $\varepsilon_s = 10^{-4}$ (f) $\varepsilon_s = 10^{-6}$

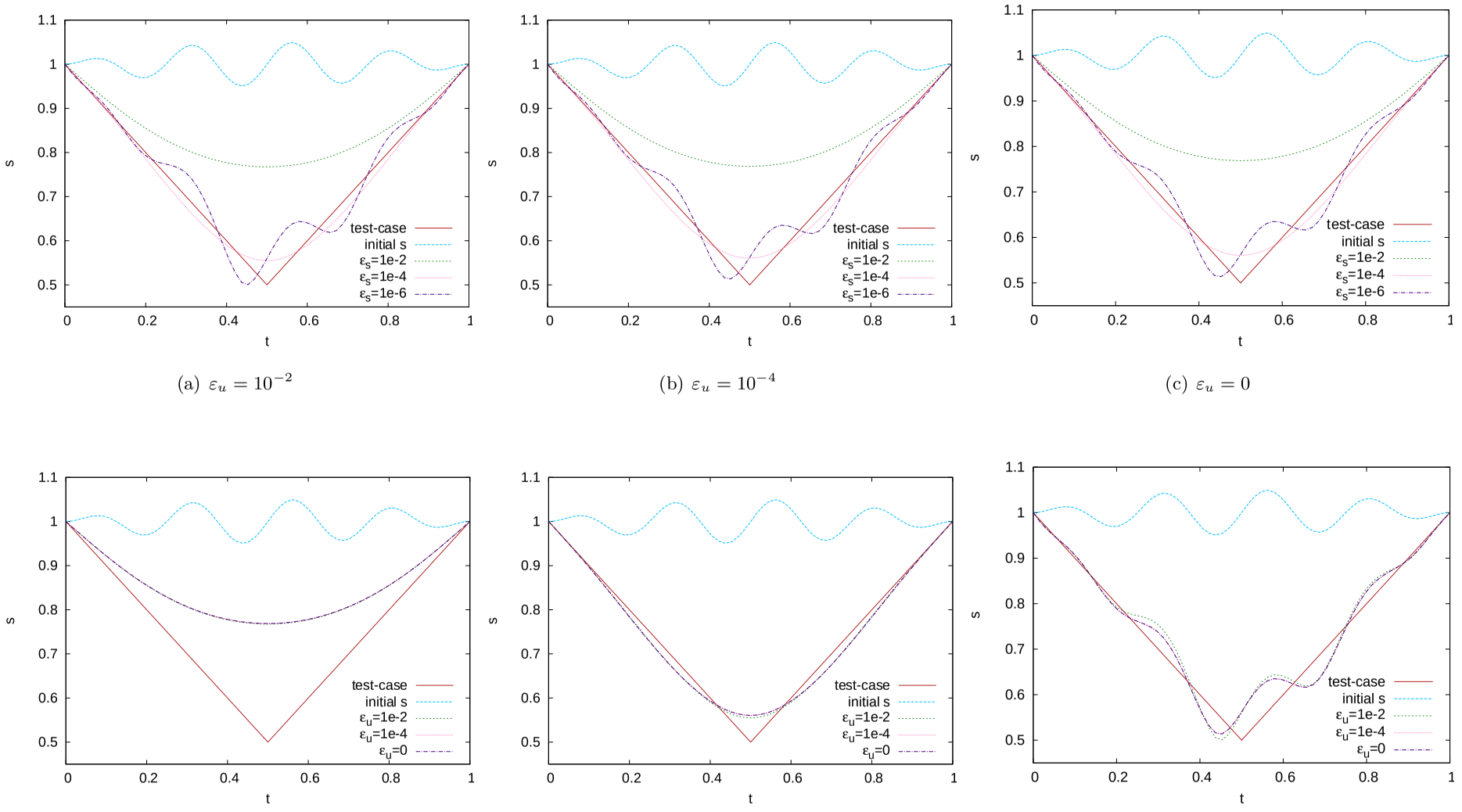
Figure B.12: Results for $s_0(t) = s_{ex}(t) + 0.2t(1-t)\sin(8\pi t)$ and $s_{ex}(t) = 1 - 5t^3(1-t)$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.6.



(a) $\varepsilon_u = 10^{-2}$ (b) $\varepsilon_u = 10^{-4}$ (c) $\varepsilon_u = 0$

(d) $\varepsilon_s = 10^{-2}$ (e) $\varepsilon_s = 10^{-4}$ (f) $\varepsilon_s = 10^{-6}$

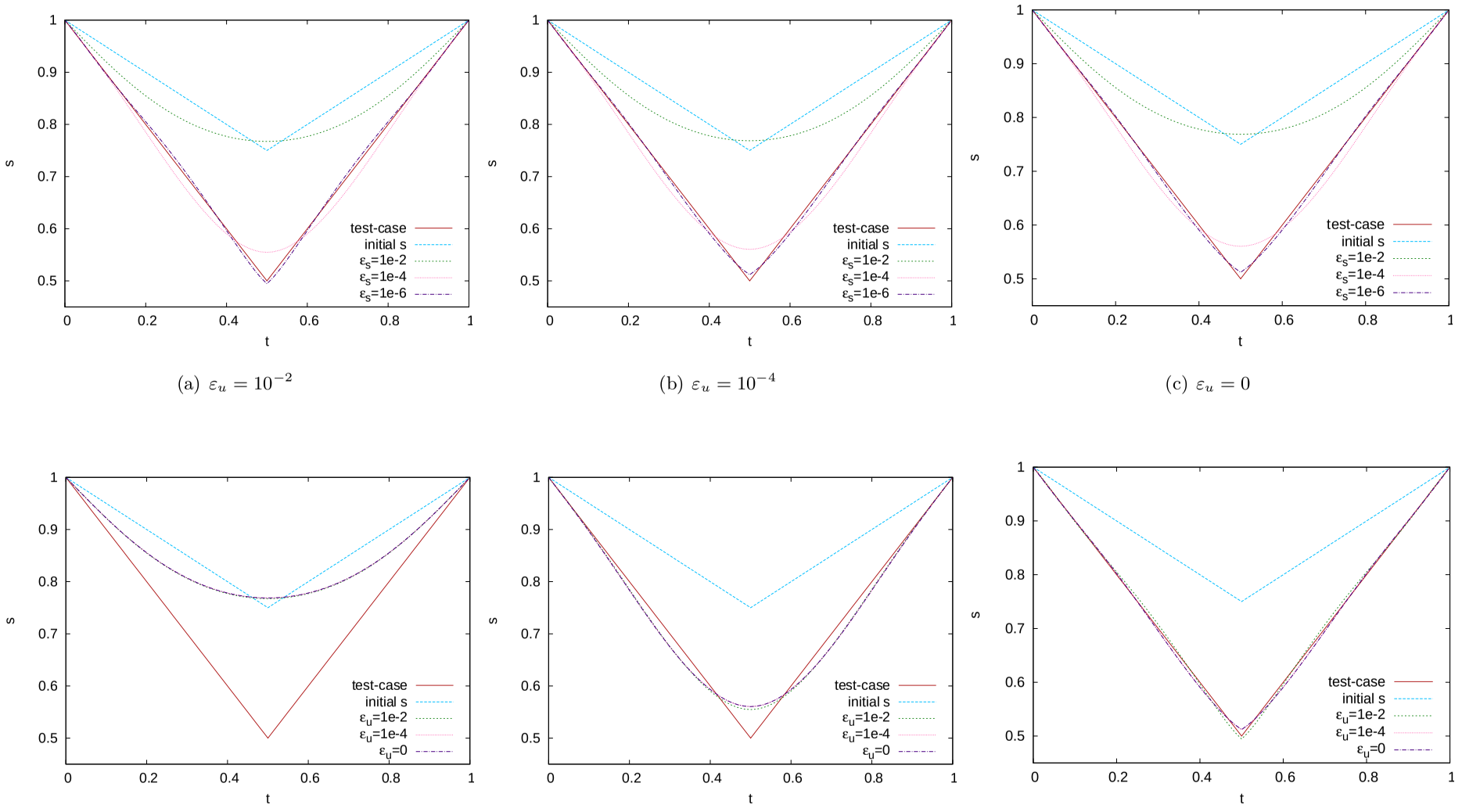
Figure B.13: Results for $s_0(t) = 1$ and $s_{ex}(t) = (1 + |2t - 1|)/2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.7.



(a) $\varepsilon_u = 10^{-2}$ (b) $\varepsilon_u = 10^{-4}$ (c) $\varepsilon_u = 0$

(d) $\varepsilon_s = 10^{-2}$ (e) $\varepsilon_s = 10^{-4}$ (f) $\varepsilon_s = 10^{-6}$

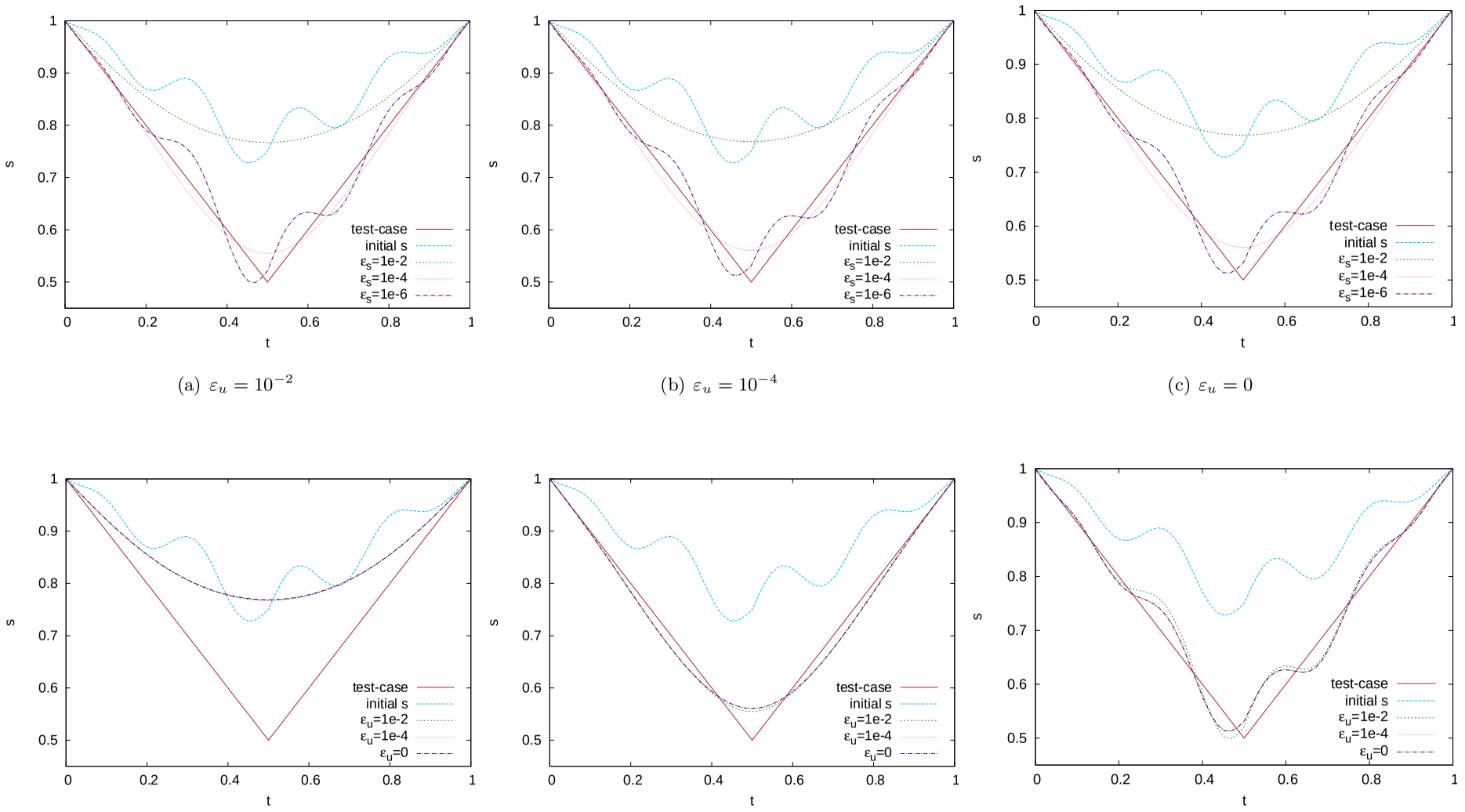
Figure B.14: Results for $s_0(t) = 1 + 0.2t(1-t)\sin(8\pi t)$ and $s_{ex}(t) = (1 + |2t-1|)/2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.7.



(a) $\varepsilon_u = 10^{-2}$ (b) $\varepsilon_u = 10^{-4}$ (c) $\varepsilon_u = 0$

(d) $\varepsilon_s = 10^{-2}$ (e) $\varepsilon_s = 10^{-4}$ (f) $\varepsilon_s = 10^{-6}$

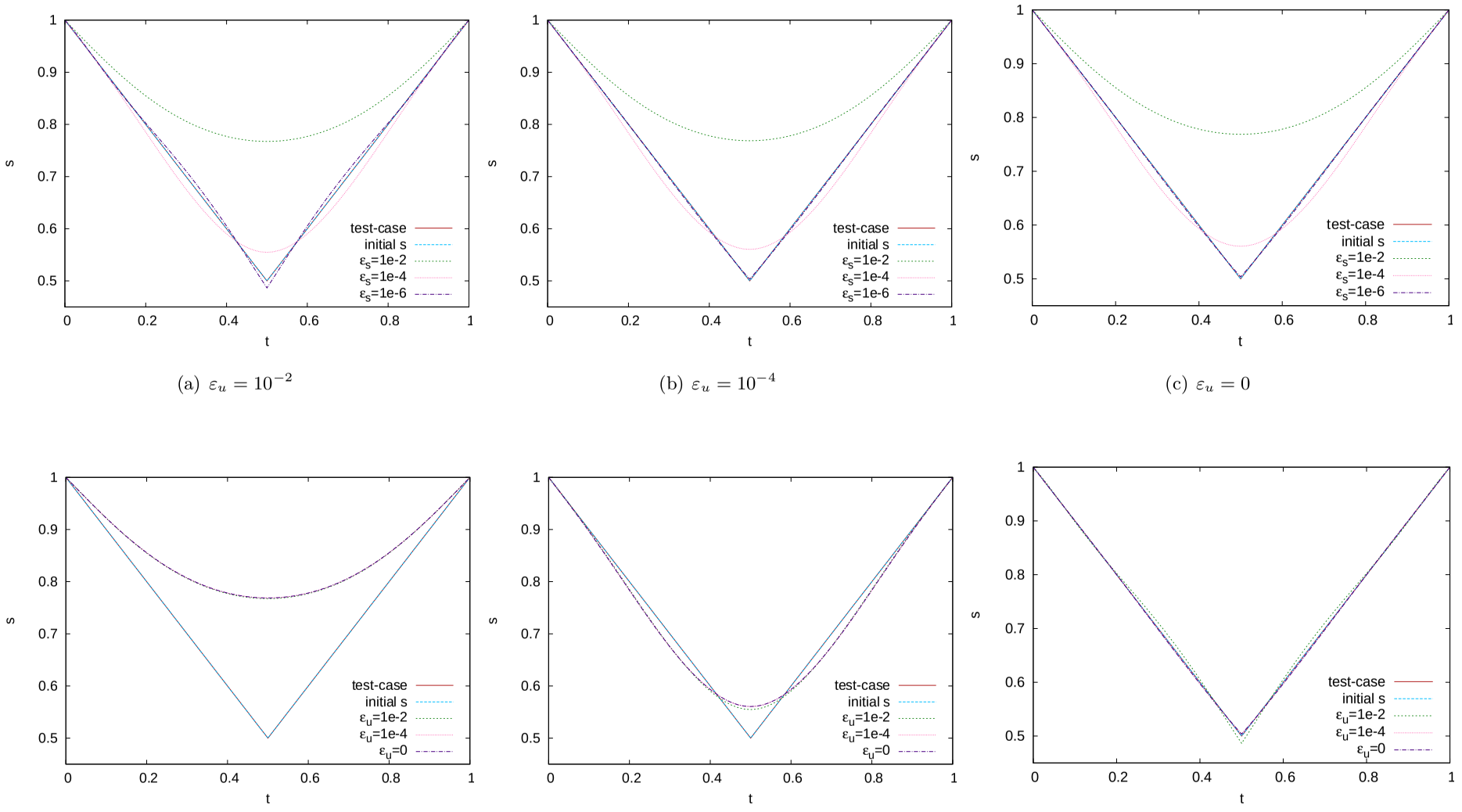
Figure B.15: Results for $s_0(t) = 0.5 + 0.5 s_{ex}(t)$ and $s_{ex}(t) = (1 + |2t - 1|)/2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.8.



(a) $\varepsilon_u = 10^{-2}$ (b) $\varepsilon_u = 10^{-4}$ (c) $\varepsilon_u = 0$

(d) $\varepsilon_s = 10^{-2}$ (e) $\varepsilon_s = 10^{-4}$ (f) $\varepsilon_s = 10^{-6}$

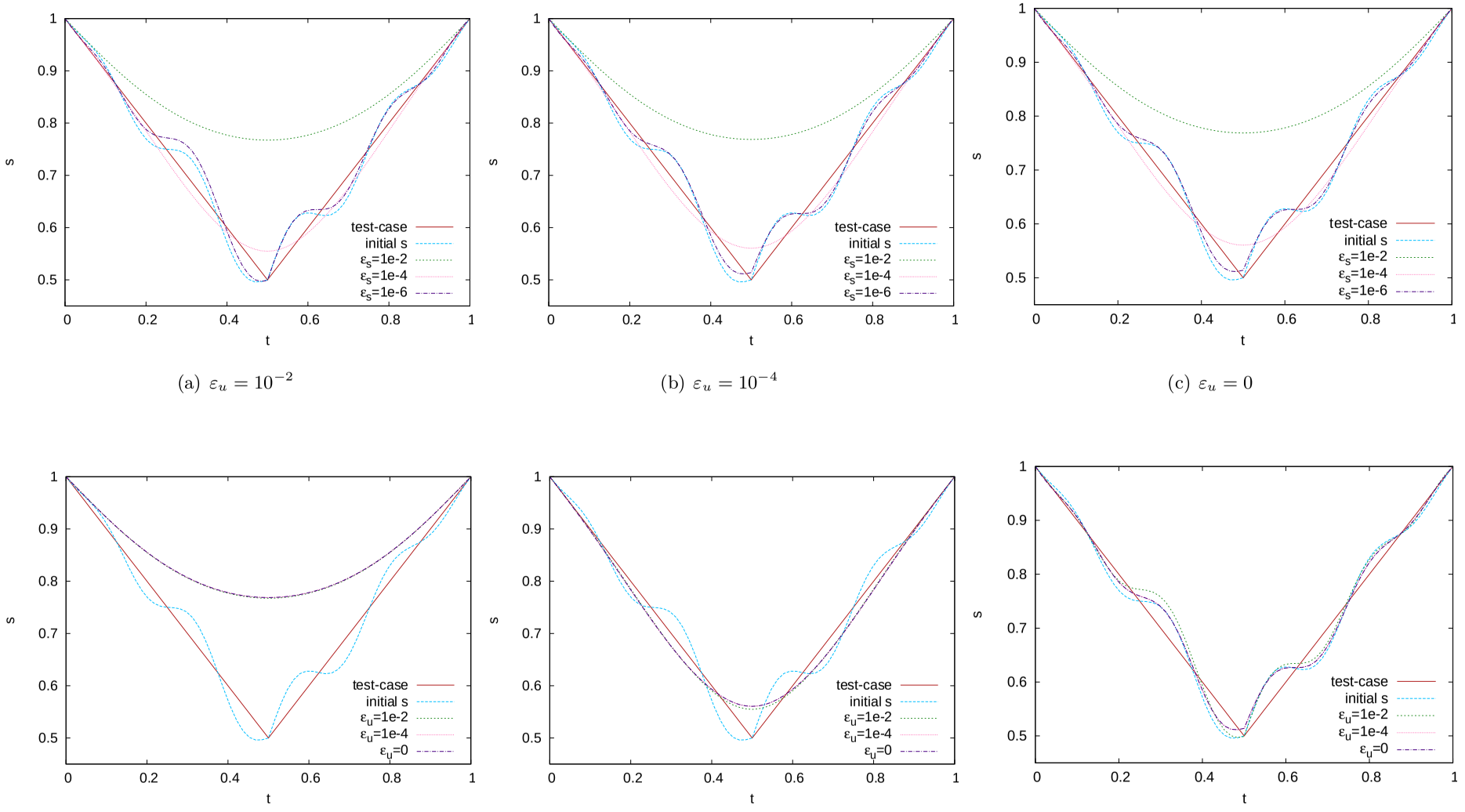
Figure B.16: Results for $s_0(t) = 0.5 + 0.5 s_{ex}(t) + 0.2t(1-t)\sin(8pi * t)$ and $s_{ex}(t) = (1 + |2t-1|)/2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.8.



(a) $\varepsilon_u = 10^{-2}$ (b) $\varepsilon_u = 10^{-4}$ (c) $\varepsilon_u = 0$

(d) $\varepsilon_s = 10^{-2}$ (e) $\varepsilon_s = 10^{-4}$ (f) $\varepsilon_s = 10^{-6}$

Figure B.17: Results for $s_0(t) = s_{ex}(t)$ and $s_{ex}(t) = (1 + |2t - 1|)/2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.9.



(a) $\varepsilon_u = 10^{-2}$ (b) $\varepsilon_u = 10^{-4}$ (c) $\varepsilon_u = 0$

(d) $\varepsilon_s = 10^{-2}$ (e) $\varepsilon_s = 10^{-4}$ (f) $\varepsilon_s = 10^{-6}$

Figure B.18: Results for $s_0(t) = s_{ex}(t) + 0.2t(1-t)\sin(8\pi t)$ and $s_{ex}(t) = (1+|2t-1|)/2$: (top row) changing ε_s; (bottom row) changing ε_u. Numbers shown in Table 4.9.

Bibliography

Ait-Oufella, H., Taleb, S., Mallat, Z. et al. (2011), 'Recent advances on the role of cytokines in atherosclerosis', *Arteriosclerosis, Thrombosis, and Vascular Biology* **31**, 969–979.

Akkus, Z., Carvalho, D. D. B., van den Oord, S. C. H. et al. (2015), 'Fully automated carotid plaque segmentation in combined contrast-enhanced and b-mode ultrasound', *Ultrasound in Medicine & Biology* **41**(2), 517531.

Alexandrov, A. V., Vital, D., Brodie, D. S. et al. (1997), 'Grading carotid stenosis with ultrasound: An interlaboratory comparison', *Stroke* **28**, 1208–1210.

Armstrong, P. A. & Bandyk, D. F. (2010), *Rutherford's Vasular Surgery*, Saunders, chapter 15, pp. 235–255.

Ascher, U. M., Ruuth, S. J. & Wetton, B. T. R. (1995), 'Implicit-explicit methods for time-dependent partial differential equations', *SIAM Journal on Numerical Analysis* **32**(3), 797–823.

Atkinson, K. (1989), *Introduction to Numerical Analysis*, 2e edn, John Wiley & Sons.

Auer, M. & Gasser, T. C. (2010), 'Automatic reconstruction and finite element generation of abdominal aortic aneurysms', *IEEE Trans Med Imag* **29**, 1022–1028.

Balla, R. Y., Stowersa, E. C., Burtona, J. H. et al. (1995), 'Evidence that the death of macrophage foam cells contributes to the lipid core of atheroma', *Atherosclerosis* **114**, 45–54.

Bangerth, W., Heister, T., Heltai, L. et al. (2013), 'The `deal.ii` library, version 8.1', *arXiv* p. 1312.2266.

Barreira, R., Elliott, C. M. & Madzvamuse, A. (2011), 'The surface nite element method

for pattern formation on evolving biological surfaces', *Journal of Mathematical Biology* **63**, 10951119.

Batchelor, G. K. (2000), *An Introduction to Fluid Dynamics*, Cambridge University Press.

Bernfield, M., Götte, M., Park, P. W. et al. (1999), 'Functions of cell surface heparan sulfate proteoglycans', *Annual Review of Biochemistry* **68**, 729–777.

Blankensteijn, J. D. & Kool, L. J. S. (2010), *Computed Tomography*, Sa, chapter 21, pp. 329–344.

Bobryshev, Y. V. (2006), 'Monocyte recruitment and foam cell formation in atherosclerosis', *Micron* **37**(3), 208–222.

Bots, M. L., Hoes, A. W., Koudstaal, P. J. et al. (1997), 'Common carotid intima-media thickness and risk of stroke and myocardial infarction: The Rotterdam study', *Circulation* **96**, 1432–1437.

Boyle, J. J. (2005), 'Macrophage activation in atherosclerosis: Pathogenesis and pharmacology of plaque rupture', *Current Vascular Pharmacology* **3**(1), 63–68.

Britz, D., Østerby, O. & Strutwolf, J. (2003), 'Damping of Crank-Nicolson error oscillations', *Computational Biology and Chemistry* **27**(3), 253–263.

Broekhuizena, L. N., Mooija, H. L., Kasteleina, J. J. et al. (2009), 'Endothelial glycocalyx as potential diagnostic and therapeutic target in cardiovascular disease', *Current Opinion in Lipidology* **20**, 57–62.

Burckhardt, C. B. (1978), 'Speckle in ultrasound B-mode scans', *Sonics and Ultrasonics, IEEE Transactions* **25**(1), 1–6.

Burke, A. P., Farb, A., Malcom, G. T. et al. (1997), 'Coronary risk factors and plaque morphology in men with coronary disease who died suddenly', *N Engl J Med.* **336**, 12761282.

Burke, G. L., Evans, G. W., Riley, W. A. et al. (1995), 'Arterial wall thickness is associated with prevalent cardiovascular disease in middle-aged adults: The atherosclerosis risk in communities (aric) study', *Stroke* **26**, 386–391.

Cai, H. & Harrison, D. G. (2000), 'Endothelial dysfunction in cardiovascular diseases: The role of oxidant stress', *Circulation Research* **87**, 840–844.

Calvez, V., Houot, J. G., Meunier, N. et al. (2010), 'Mathematical and numerical modeling of early atherosclerotic lesions', *ESAIM: Proceedings* **30**, 1–14.

Cañavate-Grimal, A., Falcó, A., Calderón, P. et al. (2015), 'On the use of stochastic spectral methods in deep excavation inverse problems', *Computers & Structures* **159**(15), 4160.

Carr, S. C., Farb, A., Pearce, W. H. et al. (1997), 'Activated inflammatory cells are associated with plaque rupture in carotid artery stenosis', *Surgery* **122**(4), 757764.

Castets, V., Dulos, E., Boissonade, J. et al. (1990), 'Experimental evidence of a sustained turingtype equilibrium chemical pattern', *Physical Review Letters* **64**(3), 29532956.

Chechkin, A., Zaid, I., Lomholt, M. et al. (2012), 'Bulk-mediated diffusion on a planar surface: Full solution', *Phys. Rev. E* **86**, 041101.

Cheng, J., Li, H., Xiao, F. et al. (2013), 'Fully automatic plaque segmentation in 3-D carotid ultrasound images', *Ultrasound in Medicine & Biology* **39**(12), 24312446.

Cilla, M., Peña, E. & Martínez, M. A. (2014), 'Mathematical modelling of atheroma plaque formation and development in coronary arteries', *J. R. Soc. Interface* **11**(90), 20130866.

Clermonta, G. & Zenke, S. (2015), 'The inverse problem in mathematical biology', *Mathematical Biosciences* **260**, 1115.

Cobbold, C. A., Sherratt, J. A. & Maxwell, S. R. J. (2002), 'Lipoprotein oxidation and its significance for atherosclerosis: a mathermatical approach', *Bulletin of Mathmatical Biology* **64**, 65–95.

Damerjian, V., Tankyevych, O., Souag, N. et al. (2014), 'Speckle characterization methods in ultrasound images a review', *IRBM* **35**(4), 202–213.

Davies, M. G. & Hagen, P.-O. (1993), 'The vascular endothelium: A new horizon', *Annals of Surgery* **218**(5), 593–609.

Davignon, J. & Ganz, P. (2004), 'Role of endothelial dysfunction in atherosclerosis', *Circulation* **109**, III–27–III–32.

Doran, A. C., Meller, N. & McNamara, C. A. (2008), 'Role of smooth muscle cells in the initiation and early progression of atherosclerosis', *Arteriosclerosis, Thrombosis, and Vascular Biology* **28**, 812–819.

Dufiet, V. & Boissonade, J. (1992), 'Conventional and unconventional turing patterns', *Journal of Chemical Physics* **96**(1), 664–673.

Dziuk, G. & Elliott, C. (2007), 'Surface finite elements for parabolic equations', *J. Comp. Math.* **25**, 385–407.

ECST (1998), 'Randomised trial of endarterectomy for recently symptomatic carotid stenosis: final results of the mrc european carotid surgery trial (ecst)', *Lancet* **351**, 1379–87.

El-said, S. A. & Azar, A. T. (2012), 'Speckles suppression techniques for ultrasound images', *Journal of Medical Imaging and Radiation Sciences* **43**(4), 200–213.

Elliott, C. & Ranner, T. (2012), 'Finite element analysis for a coupled bulk-surface partial differential equation', *IMA J. Num. Anal.* .

Elliott, C., Stinner, B. & Venkataraman, C. (2013), 'Modelling cell motility and chemotaxis with evolving surface finite elements', *J. Roy. Soc. Inter.* **9**(76), 3027–3044.

Fabbri, A. & Cevoli, C. (2016), 'Rheological parameters estimation of non-newtonian food fluids by finite elements model inversion', *Journal of Food Engineering* **169**, 172178.

Faita, F., Gemignani, V., Bianchini, E. et al. (2008), 'Real-time measurement system for evaluation of the carotid intima-media thickness with a robust edge operator', *J. Ultrasound Med.* **27**(9), 13531361.

Filipovic, N., Rosic, M., Tanaskovic, I. et al. (2011), 'ARTreat project: Three-dimensional numerical simulation of plaque formation and development in the arteries', *IEEE Transaction on Infomation Technology in Biomedecine* **(Ahead of Print)**.

Finn, A. V., Nakano, M., Narula, J. et al. (2010), 'Concept of vulnerable/unstable plaque', *Arteriosclerosis, Thrombosis, and Vascular Biology* **30**, 1282–1292.

Fok, P.-W. (2011), 'Growth of necrotic cores in atherosclerotic plaque', *Mathematical Medicine and Biology* **(Ahead of Print)**.

Formaggia, L., Quarteroni, A. & Veneziani, A., eds (2009), *Cardiovascular Mathematics : Modeling and simulation of the circulatory system*, Vol. 1 of *MS&A*, Springer.

Foteinos, G., Afzal, A. R., Mandal, K. et al. (2005), 'Heat shock protein 60 autoantibodies induce atherosclerosis apolipoprotein E - deficient mice via endothelial damage', *Circulation* **112**, 1206–1213.

Gasser, T. C., Auer, M., Labruto, F. & others. (2010), 'Biomechanical rupture risk assessment of abdominal aortic aneurysms. model complexity versus predictability of finite element simulations', *Eur J Vasc Endovasc Surg,* **40**, 176–185.

Gasser, T. C., Görgülü, G., Folkesson, M. et al. (2008), 'Failure properties of intra-luminal thrombus in abdominal aortic aneurysm under static and pulsating mechanical loads', *J Vasc Surg* **48**, 179–188.

Geest, J. P. V., Wang, D. H., Wisniewski, S. R. et al. (2006), 'Towards a noninvasive method for determination of patient-specific wall strength distribution in abdominal aortic aneurysms', *Ann Biomed Eng* **34**, 1098–116.

George, U. Z., Stephanou, A. & Madzvamuse, A. (2013), 'Mathematical modelling and numerical simulations of actin dynamics in the eukaryotic cell', *J. Math. Biol* **66**(3), 547–593.

Gerrity, R. G. (1981), 'The role of the monocyte in atherogenesis: Ii. migration of foam cells from atherosclerotic lesions', *The American Journal of Pathology* **103**(2), 191–200.

Gierer, A. & Meinhardt, H. (1972), 'A theory of biological pattern formation', *Biological Cybernetics* **12**(1), 30–39.

Glowinski, R. (2003), *Finite Element Methods for Incompressible Viscous Flow*, Vol. 9 of *Handook of Numerical Analysis*, Elsevier Science Ltd.

Gnanavel, S., Balan, N. B. & Balachandran, K. (2013), 'Simultaneous identification of parameters and initial datum of reaction diffusion system by optimization method', *Applied Mathematical Modelling* **37**(1617), 82518263.

Goldstein, J. L. & Brown, M. S. (1977), 'The low-density lipoprotein pathway and its relation to atherosclerosis', *Annual Review of Biochemistry* **46**, 897–930.

Grundtman, C., Kreutmayer, S. B., Almanzar, G. et al. (2011), 'Heat shock protein 60 and immune inflammatory responses in atherosclerosis', *Arteriosclerosis, Thrombosis, and Vascular Biology* **31**, 960–968.

Grundtman, C. & Wick, G. (2011), 'The autoimmune concept of atherosclerosis', *Current Opinion in Lipidology* **22**, 000–000.

Hairer, E., Norsett, S. P. & Wanner, G. (1987), *Solving Ordinary Differential Equations I*, Springer-Verlag.

Hegyi, L., Skepper, J. N., Cary, N. R. B. et al. (1996), 'Foam cell apoptosis and the development of the lipid core of human atherosclerosis', *The Journal of Pathology* **180**, 423–429.

Hevonoja, T., Pentikäinen, M. O., Hyvönen, M. T. et al. (2000), 'Structure of low density lipoprotein (ldl) particles: Basis for understanding molecular changes in modified ldl', *Biochimica et Biophysica Acta (BBA) - Molecular and Cell Biology of Lipids* **1488**(3), 189–210.

Holzapfel, G. A. & Ogden, R. W. (2010), 'Constitutive modelling of arteries', *Proceedings of the Royal Society A* **466**, 1551–1597.

Hughes, T. J. R. (2003), *The Finite Element Method*, Dover Publications.

Humphrey, J. D. (2003), 'Continuum biomechanics of soft biological tissues', *Proceeding of the Royal Society London A* **459**, 3–46.

Ibragimov, A. I., McNeal, C. J., Ritter, L. R. et al. (2005), 'A mathematical model of atherogenesis as an inflammatory response', *Mathematical Medicine and Biology* **22**, 305–333.

Idier, J., ed. (2008), *Bayesian Approach to Inverse Problems (ISTE)*, ISTE Ltd.

Isakov, V. (2005), *Inverse Problems for Partial Differential Equations*, Springer.

Jiang, L. & Tao, Y. (2001), 'Identifying the volatility of underlying assets from option prices', *Inverse Problems* **17**, 137155.

Kaipio, J. & Somersalo, E. (2004), *Statistical and Computational Inverse Problems*, Vol. 160 of *Applied Mathematical Sciences*, Springer.

Kasumba, H. & Kunisch, K. (2012), 'On free surface PDE constrained shape optimization problems', *Applied Mathematics and Computation* **218**, 1142911450.

Kedem, O. & Katchalsky, A. (1958), 'Thermodynamic analysis of the permeability of biological membranes to non-electrolytes', *Biochimica et Biophysica Acta* **27**, 229–246.

Khatib, N. E., Génieys, S., Kazmierczak, B. et al. (2009), 'Mathematical modelling of atherosclerosis as an inflammatory disease', *Phil. Trans. R. Soc. A* **376**, 4877–4886.

Klinke, D. J. & Birtwistle, M. R. (2015), 'In silico model-based inference: an emerging approach for inverse problems in engineering better medicines', *Current Opinion in Chemical Engineering* **10**(1424).

Kockx, M. M. & Hermanb, A. G. (2000), 'Apoptosis in atherosclerosis: beneficial or detrimental?', *Cardiovascular Reasearch* **45**, 736746.

Kolodgie, F. D., Burke, A. P., Nakazawa, G. et al. (2007), 'Is pathologic intimal thickening the key to understanding early plaque progression in human atherosclerotic disease?', *Arteriosclerosis, Thrombosis, and Vascular Biology* **27**, 986–989.

Lakkis, O., Madzvamuse, A. & Venkataraman, C. (2013), 'Implicit-explicit time-stepping with finite element approximation of RDEs on evolving domains', *SINUM* **51**, 2309–2330.

Levine, H. & Rappel, W. J. (2005), 'Membrane-bound Turing patterns', *Phys. Rev. E* **72**(6).

Libby, P., Ridker, P. M. & Maseri, A. (2002), 'Inflammation and atherosclerosis', *Circulation* **105**, 1135–1143.

Liguori, C., Paolillo, A. & Pietrosanto, A. (2001), 'An automatic measurement system for the evaluation of carotid intima-media thickness', *IEEE Trans. Instrum. Meas.* **50**(6), 16841691.

Litt, H. & Carpenter, J. P. (2010), *Magnetic Resonance Imaging*, Saunders, chapter 22, pp. 345–370.

Liu, X., Fan, Y., Deng, X. et al. (2011), 'Effect of non-Newtonian and pulsatile blood flow on mass transport in the human aorta', *Journal of Biomechanics* **44**, 1123–1131.

Macdonald, C., Merrimanb, B. & Ruuth, S. J. (2013), 'Simple computation of reaction-diffusion processes on point clouds', *Proc. Nat. Acad. Sci.* **110**, 9209–9214.

Madamanchi, N. R., Vendrov, A. & Runge, M. S. (2005), 'Oxidative stress and vascular disease', *Arteriosclerosis, Thrombosis, and Vascular Biology* **25**, 29–38.

Madzvamuse, A. (2000), A Numerical Approach to the Study of Spatial Pattern Formation, PhD thesis, University of Oxford.

Madzvamuse, A. (2006), 'Time-stepping schemes for moving finite elements applied to reaction-diffusion systems on fixed and growing domains', *Journal of Computational Physics* **214**, 239–263.

Madzvamuse, A. & Chung, A. H. W. (2014*a*), 'The bulk-surface finite element method for reaction-diffusion systems on stationary volumes', *SIAM SISC* . Under review.

Madzvamuse, A. & Chung, A. H. W. (2014*b*), 'Fully implicit time-stepping schemes and non-linear solvers for systems of reaction-diffusion equations', *Appl. Math. Comp.* **244**, 361–374.

Madzvamuse, A. & George, U. Z. (2013), 'The moving grid finite element method applied to cell movement and deformation', *Finite Element in Analysis and Design* **74**, 74–92.

Madzvamuse, A., Maini, P. K. & Wathen, A. J. (2003), 'A moving grid finite element method applied to a model biological pattern generator', *Journal of Computational Physics* **190**, 478500.

Maini, P. K., Baker, R. E. & Chong, C. M. (2006), 'The turing model comes of molecular age, (invited perspective)', *Science* **314**, 13971398.

Matsuura, E., Hughes, G. R. V. & Khamashta, M. A. (2008), 'Oxidation of ldl and its clinical implication', *Redox and Autoimmunity* **7**(7), 558–566.

Meadb, G. E., Lewisa, S. C. & Wardlaw, J. M. (2000), 'Variability in doppler ultrasound influences referral of patients for carotid surgery', *European Journal of Ultrasound* **12**(2), 137143.

Medvdev, E. S. & Stuchebrukhov, A. A. (2013), 'Mechanism of long-range proton translocation along biological membranes', *FEBS Letters* **587**, 345–349.

Mehta, T. A., Greenman, J., Ettelaie, C. et al. (2005), 'Heat shock proteins in vascular diseasea review', *European Jounal Vascular and Endovascular Surgery* **29**, 395402.

Menchón-Lara, R.-M. & Sancho-Gómez, J.-L. (2015), 'Fully automatic segmentation of ultrasound common carotid artery images based on machine learning', *Neurocomputing* **151**(1), 161167.

Michel, J.-B., Virmani, R., Arbustini, E. et al. (2011), 'Intraplaque haemorrhages as the trigger of plaque vulnerability', *European Heart Journal* **32**, 1977–1985.

Michel, L., Picasso, M., Farinotti, D. et al. (2014), 'Estimating the ice thickness of shallow glaciers from surface topography and mass-balance data with a shape optimization algorithm', *Computers & Geosciences* **66**, 182199.

Mikkonen, R., Kreula, J. M. & Virkkunen, P. J. (1996), 'Reproducibility of doppler ultrasound measurements', *Acta Radiol.* **37**, 54550.

Mochizuki, S., Vink, H., Hiramatsu, O. et al. (2002), 'Role of hyaluronic acid glycosaminoglycans in shear-induced endothelium-derived nitric oxide release', *American Journal of Physiology - Heart and Circulatory Physiology* **285**, H722–H726.

Molinari, F., Meiburger, K. M., Saba, L. et al. (2012), 'Ultrasound IMT measurement on a multi-ethnic and multi-institutional database: Our review and experience using four fully automated and one semi-automated methods', *Computer Methods and Programs in Biomedicine* **108**(3), 946960.

Molinari, F., Zengb, G. & Suri, J. S. (2010), 'A state of the art review on intimamedia thickness (IMT) measurement and wall segmentation techniques for carotid ultrasound', *Computer Methods and Programs in Biomedicine* **100**(3), 201221.

Murray, J. D. (2003), *Mathematical Biology*, Vol. 2, 3 edn, Springer.

Naghavi, M., Libby, P., Falk, E. et al. (2003*a*), 'From vulnerable plaque to vulnerable patient a call for new definitions and risk assessment strategies: Part i', *Circulation* **108**, 1664–1672.

Naghavi, M., Libby, P., Falk, E. et al. (2003*b*), 'From vulnerable plaque to vulnerable patient a call for new definitions and risk assessment strategies: Part ii', *Circulation* **108**, 1772–1778.

Nandaa, B., Maitya, D. & Maiti, D. K. (2014), 'Modal parameter based inverse approach for structural joint damage assessment using unified particle swarm optimization', *Applied Mathematics and Computation* **242**, 407422.

NASCET (1991), 'North american symptomatic carotid endarterectomy trial. methods, patient characteristics, and progress', *Stroke* **22**(6), 711–20.

Naylor, A. R. (2012), *Ultrasound and Carotid Bifurcation Atherosclerosis*, Springer, chapter 4: The Problem with Asymptomatic Carotid Stenosis, pp. 53–63.

Neilson, M., Mackenzie, J., Webb, S. et al. (2011), 'Modelling cell movement and chemotaxis using pseudopod-based feedback', *SIAM J. Sci. Comp.* **33**(3), 1035–1057.

Neilson, M. P., Veltman, D. M. & van Haaster others, P. J. M. (2011), 'Chemotaxis: a feedback-based computational model robustly predicts multiple aspects of real cell behaviour', *PLoS Biology* **9**, e1000618.

Newby, A. C. (2006), 'Matrix metalloproteinases regulate migration, proliferation, and death of vascular smooth muscle cells by degrading matrix and non-matrix substrates', *Cardiovascular Reasearch* **69**, 614–624.

Nisbet, D. R., Rodda, A. E., Finkelstein, D. I. et al. (2009), 'Surface and bulk characterisation of electrospun membranes: Problems and improvements', *Colloids and Surfaces B: Biointerfaces* **2009**(71), 1–12.

Noble, A., Johnson, R., Thomas, A. et al. (2010), *The Cardiovascular System: Systems of the Body Series*, Churchill Livingstone.

Noble, M. I., Drake-Holland, A. J. & Vink, H. (2008), 'Hypothesis: Arterial glycocalyx dysfunction is the first step in the atherosclerotic process', *QJM: An International Journal of Medicine* **101**, 513–518.

Nocedal, J. & Wright, S. (2006), *Numerical Optimization*, Springer Series in Operations Research and Financial Engineering, Springer.

Novak, I. L., Gao, F., Choi, Y. S. et al. (2007), 'Diffusion on a curved surface coupled to diffusion in the volume: Application to cell biology', *J. Comp. Phys.* **226**, 1271–1290.

Østerby, O. (2003), 'Five ways of reducing the Crank–Nicolson oscillations', *BIT Numerical Mathematics* **43**(4), 811–822.

Ougrinovskaia, A., Thompson, R. S. & Myerscough, M. R. (2010), 'An ODE model of early stages of atherosclerosis: Mechanisms of the inammatory response', *The Bulletin of Mathematical Biology* **72**, 1534–1561.

Ouyang, Q. & Swinney, H. L. (1991), 'Transition from a uniform state to hexagonal and striped Turing patterns', *Nature* **352**, 610612.

Pober, J. S. & Sessa, W. C. (2007), 'Evolving functions of endothelial cells in inflammation', *Nature Reviews Immunology* **7**, 803–815.

Prigogine, I. & Lefever, R. (1968*a*), 'Symmetry breaking instabilities in dissipative systems. ii', *Journal of Chemical Physics* **48**(4), 1695–1700.

Prigogine, I. & Lefever, R. (1968*b*), 'Symmetry breaking instabilities in dissipative systems. ii', *J. Chem. Phys.* **48**, 1695–1700.

Quarteroni, A., Saleri, F. & Sacco, R. (2006), *Numerical Mathematics*, Springer.

Rana, N. R. & McLafferty, R. B. (2010), *Arteriography*, Saunders, chapter 18, pp. 285–304.

Rätz, A. & Röger, M. (2012), 'Turing instabilities in a mathematical model for signaling networks', *J. Math. Biol.* **65**, 1215–1244.

Rätz, A. & Röger, M. (2014), 'Symmetry breaking in a bulk-surface reaction-diffusion model for signaling networks.', *Nonlinearity* **27**, 1805–1827.

Redgrave, J. N. E., Lovett, J. K., Gallagher, P. J. et al. (2006), 'Histological assessment of 526 symptomatic carotid plaques in relation to the nature and timing of ischemic symptoms : The Oxford Plaque Study', *Circulation* **113**, 2320–2328.

Redgrave, J. N., Gallagher, P., Lovett, J. K. et al. (2008), 'Critical cap thickness and rupture in symptomatic carotid plaques : The Oxford Plaque Study', *Stroke* **39**, 1722–1729.

Reitsma, S., Slaaf, D. W., Vink, H. et al. (2007), 'The endothelial glycocalyx: composition, functions and visualization', *Pflügers Archiv European Journal of Physiology* **454**, 345–359.

Ross, R. (1999), 'Atherosclerosis – an inflammatory disease', *New England Journal of Medicine* **340**(2), 115–126.

Rozada, I., Ruuth, S. & Ward, M. J. (2014), 'The stability of localized spot patterns for the Brusselator on the sphere', *SIADS* **13**(1), 564–627.

Ruuth, S. J. (1995), 'Implicit-explicit methods for reaction-diffusion problems in pattern formation', *J. Math. Biol* **34**(2), 148–176.

Saad, Y. (1992), *Numerical Methods for Large Eigenvalue Problems*, SIAM.

Sabaa, L., Molinarib, F., Meiburgerb, K. M. et al. (2013), 'Inter– and intra–observer variability analysis of completely automated cIMT measurement software (AtheroEdge^{TM}) and its benchmarking against commercial ultrasound scanner and expert readers', *Computers in Biology and Medicine* **43**(9), 12611272.

Schnakenberg, J. (1979), 'Simple chemical reaction systems with limit cycle behaviour', *Journal of Theoretical Biology* **81**(3), 389–400.

Shah, P. K. (2003), 'Mechanisms of plaque vulnerability and rupture', *J Am Coll Cardiol.* **41**(4s1), S15–S22.

Sick, S., Reinker, S., Timmer, J. & Schlake, T. (2006), 'WNT and DKK determine hair follicle spacing through a reaction-diffusion mechanism', *Science* **314**, 14471450.

Silvestre-Roig, C., de Winther, M. P., Weber, C. et al. (2014), 'Atherosclerotic plaque destabilization. mechanisms, models, and therapeutic strategies', *Circulation Research* **114**, 214–226.

Smoller, J. (1994), *Shock Waves and Reaction-Diffusion Equations*, 2nd edn, Kluwer Academic Publishers.

Solnica-Krezel, L. (2003), 'Vertebrate development: taming the nodal waves', *Current Biology* **13**, R7–9.

Standring, S. (2008), *Gray's Anatomy: The Anatomical Basis of Clinical Practice*, 40th ed., Expert Consult ed. edn, Churchill Livingstone (Elsevier Health Sciences).

Stary, H. C., Blankenhorn, D. H., Chandler, A. B. et al. (1992), 'A definition of the intima of human arteries and of its atherosclerosis- prone regions. a report from the Committee on Vascular Lesions of the Council on Arteriosclerosis, American Heart Association', *Arteriosclerosis, Thrombosis, and Vascular Biology* **12**, 120–134.

Stary, H. C., Chandler, A. B., Dinsmore, R. E. et al. (1995), 'A definition of advanced types of atherosclerotic lesions and a histological classification of atherosclerosis. a report from the Committee on Vascular Lesions of the Council on Arteriosclerosis, American Heart Association', *Arteriosclerosis, Thrombosis, and Vascular Biology* **92**, 1355–1374.

Stary, H. C., Chandler, A. B., Glagov, S. et al. (1994), 'A definition of initial, fatty streak, and intermediate lesions of atherosclerosis. a report from the Committee on Vascular Lesions of the Council on Arteriosclerosis, American Heart Association', *Arteriosclerosis, Thrombosis, and Vascular Biology* **14**, 840–856.

Stephen, S. L., Freestone, K. & Dunn, S. (2010), 'Scavenger receptors and their potential as therapeutic targets in the treatment of cardiovascular disease', *International Journal of Hypertension* **2010**, Article ID 646929, 21 pages.

Stoll, G. & Bendszus, M. (2006), 'Inflammation and atherosclerosis : Novel insights into plaque formation and destabilization', *Stroke* **37**, 1923–1932.

Stoneman, V. E. A. & Bennett, M. R. (2004), 'Role of apoptosis in atherosclerosis and its therapeutic implications', *Clinical Science* **107**, 343–354.

Sun, N., Wood, N. B., Hughes, A. D. et al. (2007), 'Inuence of pulsatile flow on LDL transport in the arterial wall', *Annals of Biomedical Engineering* **35**(10), 1781–1799.

Sundholma, J., Gustavssonb, T. & Sarkola, T. (2014), 'Semi-automatic border detection software for the quantification of arterial lumen, intima-media and adventitia layer thickness with very-high resolution ultrasound', *Atherosclerosis* **234**(2), 283287.

Tabas, I. (2010), 'Macrophage death and defective inflammation resolution in atherosclerosis', *Nature Reviews Immunology* **10**, 36–46.

Tarbell, J. M. (2003), 'Mass transport in arteries and the localisation of atherosclerosis', *Annual Review of Biomedical Engineering* **5**, 79–118.

Taylor, C. A. & Humphrey, J. D. (2009), 'Open problems in computational vascular biomechanics: Hemodynamics and arterial wall mechanics', *Computer Methods in Applied Mechanics and Engineering* **198**(45-46), 3514–3523.

Thomas, D. & Kernevez, J., eds (1976), *Analysis and control of immobilized enzyme systems*, Elsevier, chapter Articial enzyme membranes, transport, memory, and oscillatory phenomena, pp. 113–150.

Thomée, V. (2006), *Galerkin Finite Element Methods for Parabolic Problems*, 2nd edn, Springer.

Tikhonov, A. N. (1943), 'On the stability of inverse problems', *Dokl. Akad. Nauk SSSR* **39**(176-179), 184. In Russian.

Tikhonov, A. N. & Arsenin, V. Y. (1977), *Solutions of ill-posed problems.*, John Wiley & Sons, New York - Toronto. Transl. from Russian.

Todd, I. & Spickett, G. (2005), *Lecture Notes: Immunology*, 5th edn, Blackwell Publishing.

Toivanen, J. I., Haslingerand, J. & Mäkinen, R. A. E. (2008), 'Shape optimization of systems governed by Bernoulli free boundary problems', *Comput. Methods Appl. Mech. Engrg.* **197**, 38033815.

Tortora, G. J. & Derrickson, B. H. (2014), *Principles of Anatomy and Physiology*, John Wiley & Sons.

Turing, A. M. (1952), 'The chemical basis of morphogenesis', *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences* **641**, 33–72.

van den Oord, S. C. H., Sijbrands, E. J. G., ten Kate, G. L. et al. (2013), 'Carotid intima-media thickness for cardiovascular risk assessment: Systematic review and meta-analysis', *Atherosclerosis* **228**(1), 1–11.

van der Wal, A. C. & Becker, A. E. (1999), 'Atherosclerotic plaque rupture pathologic basis of plaque stability and instability', *Cardiovascular Research* **41**, 334344.

Veller, M. G., Fisher, C. M., Nicolaides, A. N. et al. (1993), 'Measurement of the ultrasonic intima-media complex thickness in normal subjects', *Journal of Vascular Surgery* **17**(4), 719725.

Venkataraman, C., Lakkis, O. & Madzvamuse, A. (2012), 'Global existence for semilinear reaction-diffusion systems on evolving domains', *J. Math. Biol* **64**, 41–67.

Venkataraman, C., Sekimura, T., Gaffney, E. A. et al. (2011), 'Modeling parr–mark pattern formation during the early dev. of Amago trout', *Physical Review E* **84**(4), 041923.

Virmani, R., Burke, A. P., Farb, A. et al. (2006), 'Pathology of the vulnerable plaque', *Journal of the American College of Cardiology* **47**(8), C138.

Virmani, R., Kolodgie, F. D., Burke, A. P. et al. (2000), 'Lessons from sudden coronary death, a comprehensive morphological classification scheme for atherosclerotic lesions', *Arteriosclerosis, Thrombosis, and Vascular Biology* **20**, 1262–1275.

Virmani, R., Kolodgie, F. D., Burke, A. P. et al. (2005), 'Atherosclerotic plaque progression and vulnerability to rupture : Angiogenesis as a source of intraplaque hemorrhage', *Arteriosclerosis, Thrombosis, and Vascular Biology* **25**, 2054–2061.

Volpert, V. & Petrovskii, S. (2009), 'Reactiondiffusion waves in biology', *Physics of Life Reviews* **6**(4), 267310.

Yang, Z. & Ming, X.-F. (2006), 'Recent advances in understanding endothelial dysfunction in atherosclerosis', *Clinical Medecine & Research* **4**(1), 53–65.

Yanga, Z. & Hamrick, J. M. (2005), 'Optimal control of salinity boundary condition in a tidal model using a variational inverse method', *Estuarine, Coastal and Shelf Science* **62**, 13–24.

Zenker, S., Rubin, J. & Clermont, G. (2007), 'From inverse problems in mathematical physiology to quantitative differential diagnoses', *PLoS Comput Biol* **3**(11), e204.

Zierler, R. E., ed. (2010), *Strandness's Duplex Scanning in Vascular Disorders*, Lippincott Williams and Wilkins.